CENTRAL EUROPEAN UNIVERSITY

MASTER'S THESIS

A MODEL CONFIDENCE SET EXTENSION OF GRID SEARCH IN HYPER-PARAMETER OPTIMIZATION

Author

MARCELL KUJBUS

Supervisor

DR. TAMÁS FERENCI

This thesis is submitted in fulfillment of the requirements for the degree of Master of Science in Mathematics.

MATHEMATICS DEPARTMENT

Contents

List of Figures

List of Tables

Abstract

Hyper-parameter optimization is a thoroughly investigated discipline in optimization theory, statistics and machine learning. While it is crucial to configure the hyper-parameters properly, the most widely used grid search technique has some hidden issues that impacts the efficiency of the learning algorithm. Such an issue is that the mentioned method does not take statistical significance of out-of-sample forecasting performance into consideration when choosing the hyper-parameter yielding the minimal loss. The following paper proposes an extension of this approach to make the modeling more robust to random fluctuation. The Model Confidence Set algorithm terminates at a superior set of hyper-parameters that are statistically indistinguishable at a given a significance level with respect to the loss their generated models produce. It is shown empirically and theoretically, that averaging over the predictions of such a superior set is more efficient with respect to out-of-sample loss than taking the prediction of a single realization of the "minimum-yielding" hyper-parameter. The theoretical argument is based on showing a specific example, where the limitations of the grid search method are reached. The empirical study examines the predictability of the evolution of the novel Covid-19 infectious disease in New York state, U.S.A. Extending the famous SEIR dynamic system with gamma distributed latent and infectious periods gives rise to hyper-parameter optimization in disease prediction.

Keywords: global optimization, hyper-parameter optimization, model selection, Model Confidence Set, autoregressive processes, maximum likelihood estimator, infectious disease models, SEIR model, gamma distribution

Chapter 1

Introduction

The ultimate objective of a typical learning algorithm $\mathscr{A}$ is to find a function g that minimizes some expected loss $L_g(x)$ over independent and identically distributed samples x having a directly not observable distribution G_x. $\mathscr{A}$ is a functional that maps a (train) data set X (finite set of samples from G_x) to a function g. Very often a learning algorithm produces g through the optimization of a training criterion with respect to a set of parameters θ. However, the learning algorithm itself may have additional features called hyper-parameters λ, so we are only able to produce results conditionally on it. As a result $g = g_\lambda = \mathscr{A}_\lambda(X)$ for an arbitrary training set X.

For example we can take a $(U_n)_{n\in\mathbb{N}}$ autoregressive process, meaning $U_n = \sum_{i=1}^{\lambda} \theta_i U_{n-i} + \varepsilon_n$, where $\theta_{1,\dots,\lambda} \in \mathbb{R}^+$ and for all $n \in \mathbb{N}$ ε_n is standard normally distributed and $\mathbb{E}\varepsilon_n\varepsilon_m = 0$ for all $n \neq m$. In this setup the θ vector is called the parameter vector, and the lag $\lambda \in \mathbb{N}$ is called the *hyper-parameter* of this model. The *parameters* are optimized through $\mathscr{A}_\lambda$, however λ is a predefined characteristic of the model itself. Therefore the value of the hyper-parameter has to be set before the learning process begins.

From now on it is a matter of taste how we choose λ, however we can define some rules of thumb. What we really need in practice is a way to choose λ so as to minimize generalization error $\mathbb{E}L_{\mathscr{A}_\lambda(X)}(x)$, where X is a data set from the distribution G_x and x is a realization of such a random variable. From the previous example it is easily seen that for every possible λ , $\mathscr{A}_\lambda$ might perform an inner optimization as well.

In general we assume that λ is point in the space Λ spanned by the possible hyper-parameters. The problem of finding the best fitting $\lambda \in \Lambda$ is called the problem of hyper-parameter optimization. This paper proposes a Model Confidence Set extension of the most widely applied

grid search algorithm for this difficult optimization problem, which is of great importance in the field of machine learning and statistics.

$$\boldsymbol{\lambda}^{(*)} = \underset{\boldsymbol{\lambda} \in \Lambda}{\operatorname{argmin}} \mathbb{E} L_{\mathscr{A}_{\boldsymbol{\lambda}}(X)}(x) \tag{1.0.1}$$

The map Ψ from $\boldsymbol{\lambda} \in \Lambda$ to $\mathbb{R}^+$, $\Psi(\boldsymbol{\lambda}) = L_{\mathscr{A}_{\boldsymbol{\lambda}}(X)}(x)$ is called the *hyper-parameter response function* (Bergstra & Bengio (2012)). Thus hyper-parameter optimization equals to the minimization of the expectation $\Psi(\boldsymbol{\lambda})$ over $\boldsymbol{\lambda} \in \Lambda$. This function is also called the *response surface* in experimental design literature.

In general we do not have efficient algorithms for performing the optimization implied by Equation 1.0.1. Theoretically we cannot even evaluate the expectation over G_x. Knowing in general very little about the response surface Ψ or the search space Λ, the dominant strategy for finding an appropriate $\boldsymbol{\lambda}$ is to choose some number n of trial points $P = \{\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)}, \ldots, \boldsymbol{\lambda}^{(n)}\}$, to evaluate $\Psi(\boldsymbol{\lambda})$ for each one and return that particular $\boldsymbol{\lambda}^{(i)}$ that worked best.

$$\boldsymbol{\lambda}^{(*)} = \underset{\boldsymbol{\lambda} \in \Lambda}{\operatorname{argmin}} \mathbb{E} L_{\mathscr{A}_{\boldsymbol{\lambda}}(X)}(x) = \underset{\boldsymbol{\lambda} \in \Lambda}{\operatorname{argmin}} \mathbb{E}\Psi(\boldsymbol{\lambda}) \approx \underset{\boldsymbol{\lambda} \in \{\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)}, \ldots, \boldsymbol{\lambda}^{(n)}\}}{\operatorname{argmin}} \mathbb{E}\Psi(\boldsymbol{\lambda})$$

The different algorithms differ in the way of choosing the trial points $\{\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)}, \ldots, \boldsymbol{\lambda}^{(n)}\}$.

The most widely used strategy is the grid search (e.g. Larochelle *et al.* (2007) and Hinton *et al.* (2006)) If Λ is a set indexed by K configuration variables, then the grid search requires that we choose a set of values for each variables $(S^{(1)}, \ldots, S^{(K)})$. In grid search the set of trials is formed by assembling every possible combination of values, hence $n = \prod_{i=1}^{K} |S^{(i)}|$. This product over K sets makes grid search suffer from the curse of dimensionality as the number of joint values grows exponentially with the number of hyper-parameters (Bellman (1961)).

There are several reasons why grid search prevails as the state of the art despite decades of research into global optimization (Nelder & Mead (1965); Kirkpatrick *et al.* (1983); Powell (1994); Weise (2009)) and the publishing of several hyper-parameter optimization algorithms (Nareyek (2003); Weihs *et al.* (2006); Hutter *et al.* (2011)):

- Grid search is simple to implement and parallelization is trivial
- Grid search typically finds better $\boldsymbol{\lambda}^{(*)}$ than purely manual sequential optimization

- Grid search is reliable in low dimensional spaces (1d, 2d)

In this paper I propose using the Model Confidence Set approach (MCS) pioneered by Hansen *et al.* (2011) as an extension. The method is quite similar to the grid search in a sense, that we need to predefine a set $(S^{(1)},\dots,S^{(K)})$ from the same configuration space as would be spanned by a regular grid. However, the later fully explained algorithm yields an iterative sequence of hypothesis tests, where in every iteration, the procedure sequentially prunes the worst performing hyper-parameter one by one with respect to their common user-defined loss function, until the first non-rejection takes place. Hence the initial trial set shrinks to a set where every $\lambda^{(i)}$ is statistically indistinguishable as a forecast performer at a given significance level α. With that said, the model confidence set covers the ground true hyper-parameter with probability $1-\alpha$. Hence the coverage is binomially distributed, with parameters n and $1-\alpha$, thus in expectation out of n trials, the MCS contains the true value $n\cdot(1-\alpha)$ times. Assuming the algorithm has terminated at a set $MCS = P^*$, we take all $\lambda \in P^*$ and define the loss ensemble as $L_{\mathscr{A}_{P^*}(X)}(x) = \frac{1}{|P^*|}\sum_{\lambda\in P^*} L_{\mathscr{A}_\lambda(X)}(x)$. This is the loss function produced by the MCS approach which needs to be compared with the best grid search loss.

In spite of the fact that MCS has not been used for hyper-parameter optimization, it performs remarkably well in other areas such as forecast combination. Samuels & Sekkel (2017) analyzes the effects of trimming the set of models prior to averaging their predictions[1]. They used the Model Confidence Set approach, which takes into account the statistical significance of the out-of-sample forecasting performance. In an empirical application to the forecasting of U.S. macroeconomic indicators, they find significant gains in out-of-sample forecast accuracy from using the proposed trimming method.

Bernardi & Catania (2018) have built an R package called *MCS* to illustrate how the MCS sequence of tests delivers the superior set of models having equal predictive ability in terms of a user supplied loss function discriminating models with respect to the desired model characteristics such as forecast performances. They have applied the algorithm to different models belonging to the ARCH family to predict financial losses. They found that the use of MCS remarkably improves VaR forecast performance.

There are papers that do not share the dominance of the proposed approach. Aparicio & López de Prado (2018) evaluated the performance of the Model Confidence Set in a simple machine learning trading strategy problem, as they state that most of the model selection methods available in modern finance are subject to backtest overfitting. However, they have

[1] Averaging the forecasts from a range of models often improves upon forecasts based on a single model, with equal weight averaging working particularly well (Timmermann (2006)).

found that MCS is not robust to multiple testing and that it requires a very high signal-to-noise ratio to be utilizable.

This paper is organized as follows. Chapter 2 introduces the Model Confidence Set approach forked to the objective of hyper-parameter optimization. Chapter 3 shows an encouraging example to shed light on the comparative advantage of MCS over the grid search method. That chapter assumes a very specific Data Generating Process – an autoregression – to see that models generated by the different hyper-parameters can cause very flat loss curves. The next chapter is an empirical investigation to compare how MCS and grid search works "outside the laboratory". In light of the outbreak of novel corona virus – Covid-19 – I try to build models to predict the future values of the disease. Based on Wearing *et al.* (2005a), I build a complex system of delayed differential equations with two hyper-parameters and several parameters.

Some of the upcoming figures are available interactively at *this*[2] webpage. Additionally, I have created an online appendix for the thesis. Visit the website if you would like to see how the implementation of the models happened in code. The appendix can be found by clicking *here*[3].

[2] *https://rpubs.com/kujbika*

[3] *https://github.com/kujbika/MCS_in_hyperparam_optim_APPENDIX*

Chapter 2

The Model Confidence Set in hyper-parameter optimization

2.1 Assumptions

I now introduce how the Model Confidence Set algorithm developed by Hansen *et al.* (2011) applies to hyper-parameter optimization. Let

- the trial set of hyper-parameters λ be $P = \{\lambda^{(1)}, \lambda^{(2)}, \ldots, \lambda^{(n)}\}$, where $n = \prod_{i=1}^{K} |S^{(i)}|$, where $S^{(i)}$ is the set of the i^{th} configuration variable from 1 to K.
- $(\Omega, \mathscr{F}, \mathbb{P})$ be a probability space
- $Y : \Omega \times \mathbb{R}^+ \to \mathbb{R}$ be the underlying stochastic process that we aim to model[1]
- $M : \mathscr{P}(P) \to \mathscr{P}(\mathscr{A})$ be a bijection from a set of P to a set of learning algorithms[2]. For example $M(P) = (\mathscr{A}_\lambda)_{\lambda \in P}$, $M(\lambda) = \mathscr{A}_\lambda$.
- X be the training data set, i.e. a finite set of samples from a common distribution G_x
- $\Psi : P \times \mathbb{R}^+ \to \mathbb{R}$, $\Psi(\lambda, t) = L_{M(\lambda)(X)}(t)$ be the response surface or hyper-parameter response function defined on the set of models and on the time. For example assuming L_2-norm loss $\Psi(\lambda, t) = (Y_t - g_{\mathscr{A}_\lambda(X)}(x_t))^2$, where Y_t is the value of the process at time t , X is the train set and x_t form the predictors of Y_t based on X.

[1] In other words, $Y = Y_t$ is random function, that we can only observe through random noises.
[2] $\mathscr{P}$ stands for the power set.

- $d : (P \times P) \times t$, $d_{ij}(t) = \Psi(i,t) - \Psi(j,t)$ be the difference function between the losses of the models generated by i and j at given time t.
- $\alpha \in (0,1)$ be the (fixed) significance level.

Hence the ingredients for our MCS recipe have been listed. $d_{ij}(t)$ as a $(\mathbb{R}, \mathscr{B}_{\mathbb{R}})$ measurable function is a random variable for every time t for every fixed i, j. In the followings assume, that the expectation of $d_{ij,t}$ is time-homogeneous i.e the expectation of the loss $\mathbb{E}d_{ij}(t) = \mathbb{E}d_{ij}, \forall t \in \mathbb{R}^+$ is constant throughout time, thus it depends only on i and j.

2.2 The algorithm

2.2.1 The hyper-parameters construct a well-ordered set

Definition 1. Define $\succsim \subseteq P \times P$ the relation between hyper-parameters i and j with the following properties:

$$i \succsim j \iff \mathbb{E}d_{ij} \leq 0$$

Furthermore we say, that i and j are equivalent if $i \succsim j$ and $j \succsim i$, i.e. $\mathbb{E}d_{ij} = 0$.[3]

Remark. $(P, \succsim)$ is a totally ordered set.

Indeed, $\succsim$ is antisymmetric, transitive, reflexive and for any $i, j \in P$ $i \succsim j$ *or* $j \succsim i$.

Theorem 2. $(P, \succsim)$ *is well-ordered.*

Proof. Recall that a set is well-ordered, if a total order relation is defined on it having that every non-empty subset of the set has a least-element. The least-element is necessarily unique up to equivalence. Formally let $S \subset P$, then the model generated by j is the least-element of the set S, if $i \succsim j$, $\forall i \in S$.

Suppose that $|S| = n \in \mathbb{N}$. In that case the set $d(S)$ of all the difference function values along S has n^2 elements, and $d(S) \subset \mathbb{R}$. Now we remain to prove that every finite subset of $\mathbb{R}$ is well-ordered, with the total order relation $\leq$.

The proof is on induction: Every singleton is trivially well-ordered (if the set is $\{x\}$, then $x \leq x$ is a well-order). Let's see now $A = \{r_1, r_2, ..., r_n\}$. We need that every $B \subset A$ has a least-element. Of course, $|B| < |A|$, so by the induction hypothesis A is well-ordered. □

[3] i is preferred to j if the loss assigned to i is not greater than the loss of j.

Corollary. *There exists uniquely a superior set* $P^* = \{i \in P : \ i \succsim j, \ \forall j \in P\}$.

Remark. By the uniqueness property of the proof of the above theorem, P^* consists of models, that are equivalent to each other in a sense, that for each $k \in P^*$ $\mathbb{E}\Psi(i) = \mathbb{E}\Psi(j)$. [4] Note that $\mathbb{E}d_{ij} \leq 0 \forall i \in P^*, \forall j \notin P^*$, that is the expected loss is minimal in P^* across P.

The Model Confidence Set approach aims to find P^*, without directly knowing the expectation of the d_{ij} random variables. For the procedure, only a sample of the losses are observable.

2.2.2 Hypothesis testing

For an arbitrary subset $A \in \mathscr{P}(P)$, let the null hypothesis $H_{0,A}$ be the event that $\{\mathbb{E}d_{ij} = 0 \ \forall i, j \in A\}$. The alternative hypothesis $H_{1,A}$ is the exact complement. Introduce the *equivalence test* δ_A, that tests $H_{0,A}$ at a α significance level. As a convention, we let $\delta_A = 0$ and $\delta_A = 1$ correspond to the cases where $H_{0,A}$ are accepted and rejected, respectively. Define an *elimination rule* e_A that ranks the hyper-parameters in $(A, \overline{\succsim})$[5], and chooses the one that produces A's least-element with respect to the relation $\overline{\succsim}$(by the well-ordered principle it is always achievable). Thus the elimination rule chooses the hyper-parameter that produces the biggest expected loss.

The algorithm, visualized by Figure 2.2.1, is as follows.

1. Initially set $A = P$.
2. Test $H_{0,A}$ using δ_A at level α.
   (a) If the null hypothesis gets rejected, apply the elimination rule and construct A with throwing the least element, then repeat the procedure from Step 1.
   (b) Otherwise define $\hat{P}_{1-\alpha} = A$

The algorithm terminates at a set $\hat{P}_{1-\alpha}$, which consists of the set of surviving hyper-parameters (those that survived all tests without being eliminated), is referred to as the *model confidence set.* *Hansen* et al. *(2011)* has shown that given some condition $\hat{P}_{1-\alpha}$ consists of hyper-parameters that covers the grand true value with probability $1 - \alpha$. At theorem 3 I clarify those assumptions.

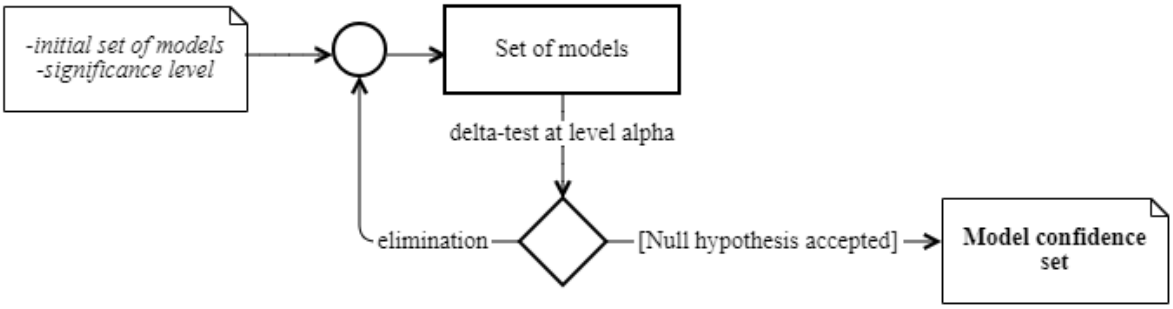
[4] In the followings, I will not write t as a parameter, as we assumed that the loss differences are time-homogeneous.

[5] $\overline{\succsim} \subseteq P \times P$, $i \overline{\succsim} j \iff \mathbb{E}d_{ij} \geq 0$

Sequential testing is key for building MCS. However, econometricians often worry about false discoveries, due to p-hacking, i.e such a testing procedure can "accumulate" Type 1 error with unfortunate consequences (for example Leeb & Pötscher (2003) or Ioannidis (2005)). The MCS procedure does not suffer from this problem because the sequential testing is halted when the first hypothesis is accepted.

Note that $\hat{P}_{1-\alpha}$ is dependent on the sample of course, and is seen through randomness. The natural question is at what extent $\hat{P}_{1-\alpha}$ reflects the actual P^*? The below results shows that the term confidence set is appropriate in this context.

Figure 2.2.1: The MCS algorithm in a UML Activity diagram



Own figure

Theorem 3. *Finite and asymptotic termination of the MCS algorithm*

If the following three conditions hold:

- $\lim_{n\to\infty} \mathbb{P}(\delta_A = 1 | H_{0,A}) \leq \alpha$, *meaning* δ *is a valid-test*
- $\limsup_{n\to\infty} \mathbb{P}(\delta_A = 1 | H_{1,A}) = 1$, *meaning* δ *has a power of 1 asymptotically*
- $\lim_{n\to\infty} \mathbb{P}(e_A \in P^* | H_{1,M}) = 0$, *meaning asymptotically any element thrown out is almost surely not in the superior set,*

then:

- $\liminf_{n\to\infty} \mathbb{P}(P^* \subset \hat{P}_{1-\alpha}) \geq 1 - \alpha$, *hence the confidence set nomenclature*
- $\lim_{n\to\infty} \mathbb{P}(i \in P^* | i \notin P^*) = 0$
- $\lim_{n\to\infty} \mathbb{P}(P^* = \hat{P}_{1-\alpha}) = 1$, *if* $|P^*| = 1$

Furthermore, if $\mathbb{P}(\delta_A = 1, e_A \in P^*) \leq \alpha$*), meaning there is coherency between the equivalence test and the elimination rule, then the main result holds on a finite sample as well:* $\mathbb{P}(P^* \subset \hat{P}_{1-\alpha}) \geq 1 - \alpha$.

You might ask at this point: do we have such a δ-test and such an elimination rule? I will return to this question at section 2.2.4.

2.2.3 The MCS p-values

In this section I introduce the notion of MCS p-values. The goal of defining such values it to study what hyper-parameters will make it into the MCS. Thus for this section, we do not terminate the algorithm at the first rejection. This is only a technical detail, the algorithm itself of course terminates at some point. However, as it turns out the MCS contains hyper-parameters that have some property in common.

The elimination rule e_A defines a sequence of random sets $P = A_1 \supset A_2 \supset \cdots \supset A_n$, where $A_i = \{e_{A_i}, \ldots, e_{A_n}\}$. So e_{A_1} is the first hyper-parameter to be eliminated in the event $H_{o,P}$ is rejected, e_{A_2} is second element to be eliminated and so forth.

Definition 4. The MCS p-values: Let $p_{H_{0,A_i}}$denote the p-value associated with the null hypothesis H_{0,A_i}, with the convention that $p_{H_{0,A_n}} \equiv 1$. The MCS p-value for model $e_{A_j} \in P$ is defined by $\hat{p}_{e_{A_j}} = \max_{i \leq j}(p_{H_{0,A_i}})$

Thus the MCS p-values form a monotonically increasing sequence. Since A_n consists of a single hyper-parameter, the null hypothesis H_{0,A_n} simply states that the last surviving model generated by that hyper-parameter is as good as itself, making the convention $p_{H_{0,A_n}} \equiv 1$ logical. Such p-values are convenient because they make is easy to determine whether a particular object is in $\hat{P}_{1-\alpha}$ for any α. Thus the MCS p-values are an effective way to convey the information in the data.

One possible realization of the MCS p-values are seen at Table 2.1.

Theorem 5. *The MCS p-value,* $\hat{p}_i$*, is such that* $i \in \hat{P}_{1-\alpha}$ *if and only if* $\hat{p}_i \geq \alpha$ *for any* $i \in P = \{\lambda^{(1)}, \lambda^{(2)}, \ldots, \lambda^{(n)}\}$.

Proof. Suppose that $\hat{p}_i < \alpha$ and determine the k for which $i = e_{A_k}$. Since $\hat{p}_i = \hat{p}_{A_k} = \max_{j \leq k}(p_{H_{0,A_j}})$, it follows that $H_{0,A_1}, \ldots, H_{0,A_k}$ are all rejected at a significance level α. Hence the first accepted hypothesis must occur after $i = e_{A_k}$has been eliminated. So $\hat{p}_i < \alpha$ implies $i \notin \hat{P}_{1-\alpha}$.

Suppose now that $\hat{p}_i = \hat{p}_{A_k} \geq \alpha$. Then for some $j \leq k$, we have $p_{H_{0,A_j}} \geq \alpha$, in which case H_{0,A_j} is accepted at significance level α that terminates the MCS procedure before the elimination rule gets applied. So $\hat{p}_i \geq \alpha$ must imply $i \in \hat{P}_{1-\alpha}$. Hence the proof. □

Table 2.1: MCS p-values

Note that the MCS p-values for some models do not coincide with the p-values for the corresponding null hypothesis. For example, the MCS p-value for e_{A_3} (the third hyper-parameter to be eliminated) exceeds the p-value for H_{0,A_3}, because the p-value associated with H_{0,A_2} - a null hypothesis tested prior to H_{0,A_3} - is larger.

Elimination rule	p-value for H_{0,A_k}	MCS p-value
$e_{A_1} = e_P$	$p_{H_{0,A_1}} = 0.01$	$\hat{p}_{e_{A_1}} = 0.01$
e_{A_2}	$p_{H_{0,A_2}} = 0.04$	$\hat{p}_{e_{A_2}} = 0.04$
e_{A_3}	$p_{H_{0,A_3}} = 0.02$	$\hat{p}_{e_{A_3}} = 0.04$
e_{A_4}	$p_{H_{0,A_4}} = 0.07$	$\hat{p}_{e_{A_4}} = 0.07$
e_{A_5}	$p_{H_{0,A_5}} = 0.05$	$\hat{p}_{e_{A_5}} = 0.07$
...	...	...
e_{A_2}	$p_{H_{0,A_n}} \equiv 1$	$\hat{p}_{e_{A_n}} = 1$

2.2.4 The existence of such a system

Hansen *et al.* (2011) has shown some particular choices of δ equivalence tests and e elimination rules that match the above mentioned criteria. Hence this section develop two tests based on the authors.

2.2.4.1 The equivalence test

Let A consists of k hyper-parameters ($k \leq n$). Define the $\bar{d_{ij}} = \frac{1}{m}\sum_{t=1}^{m} d_{ij}(t)$ and $\bar{d_i} = \frac{1}{k}\sum_{j \in A} \bar{d_{ij}}$. Here $\bar{d_{ij}}$ measures the relative sample loss between the models generated by the i^{th} and the j^{th} hyper-parameter, while $\bar{d_i}$ is the sample loss of the model generated by the i^{th} hyper-parameter relative to the average across hyper-parameters in A.

We can now construct the t-statistics $t_{ij} = \frac{\bar{d_{ij}}}{\sqrt{\widetilde{Var}(\bar{d_{ij}})}}$ and $t_i = \frac{\bar{d_i}}{\sqrt{\widetilde{Var}(\bar{d_i})}}$, where $\widetilde{Var}$ denotes the variance estimator of the respective variances. The first statistic t_{ij} is used in the famous Diebold and Mariano test (see Diebold & Mariano (1995) and West (1996)).

These statistics form the basis of $H_{0,A}$, as e.g. if for $A \subset P$, where $A = (\lambda^{(1)}, \ldots, \lambda^{(k)})$, then $H_{0,A} = \{\mathbb{E}\Psi(\lambda^{(1)}) = \mathbb{E}\Psi(\lambda^{(2)}) = \ldots = \mathbb{E}\Psi(\lambda^{(k)}) = 0\} = \{\mathbb{E}(\Psi(i) - \Psi(j)) = 0, \forall i, j \in A\}$.

Corollary. *The test statistics with the desired properties (described at Theorem 3) for $H_{0,A}$ is given by $T_{max,A} = \max_{i \in A} t_i$ and $T_{R,A} = \max_{i,j \in A} |t_{ij}|$.*

Unfortunately the distribution of these statistics are not in a closed form, as they depend on the covariance of such t-tests for example, and they are too complex in this setup. However, fortunately the relevant distributions can be estimated with bootstrap methods that implicitly deal with the nuisance parameters.

2.2.4.2 The elimination rule

Characterization of the MCS procedure needs an elimination rule that satisfies the assumptions of Theorem 3. Fortunately, such a rule comes free for the $T_{max,A}$ or the $T_{R,A}$ test statistics.

Case 1. For the test statistic $T_{max,A}$, $e_{max,A} = \underset{i \in A}{argmax}\, t_i$ is a natural elimination rule, because a rejection of the null hypothesis identifies the null $\mathbb{E}\Psi(i) = 0$ false for $i = e_{max,A}$. In this case the rule eliminates that particular model that contributes the most to the test statistic, or that particular model has the largest standardized excess loss relatively.

Case 2. On the other hand, $e_{R,A} = \underset{i \in A}{argmax} \underset{j \in A}{sup}\, t_{ij}$ is a coherent choice for the statistic $T_{R,A}$, because this model is such that $t_{e_{R,A_j}} = T_{R,A}$ for some $j \in A$

Proposition. *Let $\delta_{max,A}$ and $\delta_{R,A}$ denote the test statistics $T_{max,A}$ and $T_{R,A}$ respectively. Then $(\delta_{max,A}, e_{max,A})$ and $(\delta_{R,A}, e_{R,A})$ both satisfy the coherency assumption to make the MCS works for finite samples as well.*

Proof. 14^{th} page in the paper of Hansen *et al.* (2011). □

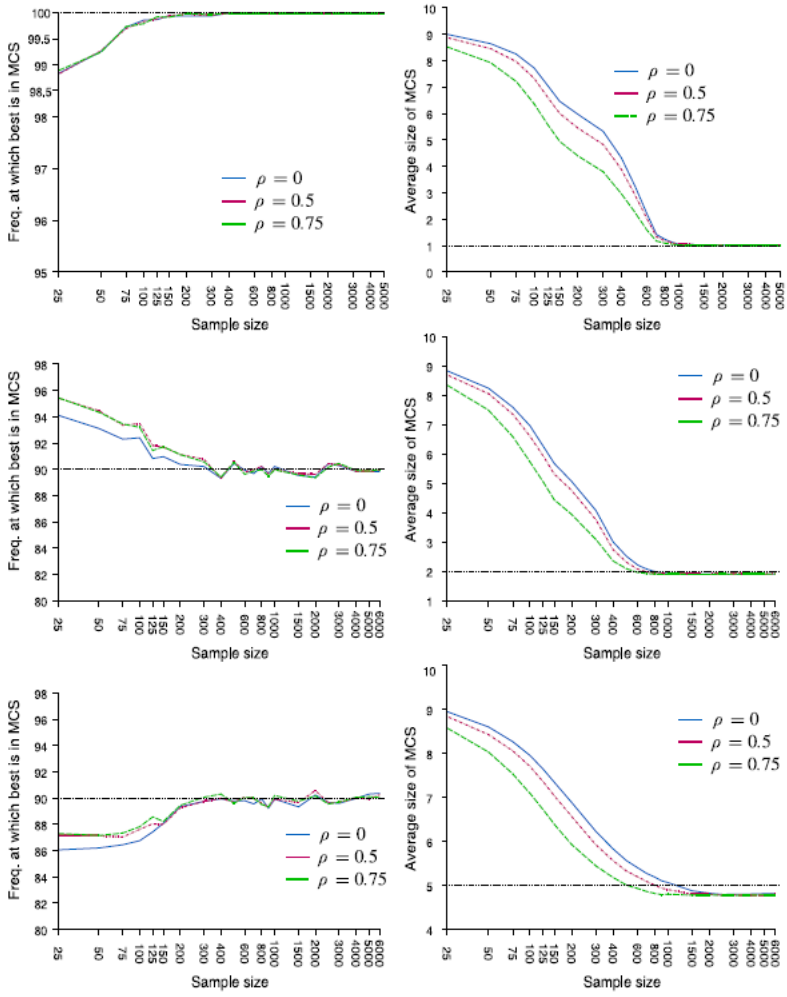
2.3 A simulation experiment

In the following I present a simulation experiment directly from the work of Hansen *et al.* (2011). The design sets the losses across models dependent, that is, $L_t \sim \mathcal{N}(\mu, \Sigma)$, where the loss vector consist of 10 elements and the covariance matrix has structure $\Sigma_{i,j} = \rho^{|i-j|}$ for three different setups: $\rho = 0, 0.5$ and 0.75. The mean vector takes the form of $\mu = (0, \ldots, 0, \frac{1}{5}, \ldots, \frac{1}{5})^T$ so that the number of zero elements define the number of elements of

the superior set P^*. They made a report showing the simulation results for the case when $|P^*| = 1, 2$ or 5. The authors used the T_{max} stat to obtain the results. The simulation results are presented at Figure 2.3.1.

Figure 2.3.1: How MCS works on dependent losses

The left panels display the frequency at which $\hat{P}_{90\%}$ contains P^*at various sample sizes. The right panel present the average number of models in $\hat{P}_{1-\alpha}$. The two upper panels contain the results for the case when $|P^*| = 1$. The panels at the middle show the results in the case $|P^*| = 2$. The two plots below describes the behaviour of the MCS procedure when $|P^*| = 5$.



Source of figure: Hansen *et al.* (2011) 28th page (Figure 1.)

Chapter 3

An encouraging example - modeling an autoregressive process

3.1 The goal of the chapter

Grid search and MCS behaves similarly in a sense, that both needs to compute losses for all possible configurations. MCS shrinks to the superior set containing the ground true values of the hyper-parameters yielding minimal losses. Grid search always chooses a singleton set that produces the minimal loss in a realization of an experiment.

When the loss realizations decrease sharply, the grid search method chooses the best hyper-parameter with a high probability. Note that in this case the MCS also chooses this hyper-parameter too as the singleton superior set. Therefore, MCS and grid search performs similarly, *unless the loss realizations are flat.* In that case - as MCS always takes into account the statistical unassertavity - MCS remarkably outperforms its competitor. Hence, the goal of this chapter is to show that such an experiment (like the one presented at section 2.3) truly exists in the ”real life”. It would imply that the use of MCS statistically dominates the use of grid search.

Fortunately, as we shall see, an autoregression fulfills all the criterion to show the usability of the proposed hyper-parameter optimization method.

3.2 An autoregressive process - assumptions

Let us take the autoregressive process as an example. That is:

1. The underlying discrete stochastic process follows an autoregressive trend, with parameter $p \in \mathbb{N}$, meaning,

$$Y_t = \alpha + \phi_1 Y_{t-1} + \cdots + \phi_p Y_{t-p} + \varepsilon_t,$$

   where $\forall t \in \mathbb{N},\ \varepsilon_t \sim \mathcal{N}(0, \sigma^2)$ and $\alpha \in \mathbb{R},\ \phi \in \mathbb{R}^p,\ \sigma^2 \in \mathbb{R}^+$ are fixed constants.

2. Suppose that the roots of the characteristic polynomial $H(z) = 1 - (\phi_1 z + \cdots + \phi_p z^p)$ lie within the unit circle and so, there exists a stationary solution Y_t of the equation. If that holds, $\lim_{t\to\infty} \mathbb{E}Y_t = \frac{\alpha}{1 - \sum_{i=1}^{p} \phi_i}$, $\lim_{t\to\infty} Var(Y_t) := D^2$.

3. For every $i \neq j$, $\mathbb{E}\varepsilon_i\varepsilon_j = 0$, so the process is homoskedastic and the error terms are uncorrelated.

Our modeling assumption are the followings:

1. There is an initial set of orders to experiment on the grid, containing the true value p: $\mathscr{P} = (p, p_1, \ldots, p_{n-1})$. $\mathscr{P}$ is the set of hyper-parameters.

2. For every $i \in \mathscr{P}$,we model the the process assuming the order of the autoregressive process is exactly i. The $(\alpha, \phi_1, \ldots, \phi_i, \sigma^2)^T$ vector is found via an inner optimization, in this case by maximizing the likelihood, that Y follows an autoregressive trend with order i.

3. The loss function is the L_2-norm.

3.3 The inner optimization - maximum-likelihood estimators

Fix $i \in \mathscr{P}$. Let n be the greatest number in $\mathscr{P}$.

Let $\theta_i = (\phi_i, \sigma_i^2)^T$ be the vector of parameters to find, where $\phi_i = [\alpha, \phi_1, \ldots, \phi_i]^T$. Given a single trajectory y of Y up to time t, which parameters maximize the likelihood that given

the process is sampled from an $AR(i)$, the actual data is observed? In other terms, which configuration of the hyper-parameter θ_i maximizes the joint density function f of y given that Y follows an $AR(i)$?

$$f_{\theta_i}(y_0, y_1, \ldots, y_i, \ldots, y_p, \ldots, y_T) = f_{\theta_i}(y_T | y_{T-1}, y_{T-2}, \ldots, y_1) f_{\theta_i}(y_1, \ldots y_{T-1}) = \cdots =$$

$$= f_{\theta_i}(y_1, \ldots, y_n) \prod_{j=n+1}^{T} f_{\theta_i}(y_j | y_{j-1}, \ldots, y_1)$$

Assuming the first n observations are deterministic, $\forall j \in 1, \ldots, T \; f_{\theta_i}(y_j | y_{j-1}, y_{j-2}, \ldots, y_1) = \begin{cases} f_{\theta_i}(y_j | y_{j-1}, \ldots, y_{j-i}), & if \; j \geq n \\ 1, & if \; j \leq n \end{cases}$. By this assumption, the joint density function to maximize is:

$$\prod_{j=n+1}^{T} f_{\theta_i}(y_j | y_{j-1}, \ldots, y_1)$$

In practice, this approach is called the conditional maximum likelihood estimation. That is, the first n observations are handled to be deterministic, thus their density must equal one. The method that takes the first n arguments as random variables is called the exact maximum likelihood estimation. However, the solution of the exact one is not closed for any ordered autoregressive process (Miller (1995)). In the scope of the current paper we aim to obtain analytically tractable results, thus we apply the conditional ML approach.

Furthermore, by the properties of an autoregressive stochastic process $y_j | y_{j-1}, \ldots, y_{j-i} \sim \mathcal{N}(x_i^T \phi_i, \sigma^2)$. This yields given that $x_{t,i} = [1, y_{t-1}, \ldots, y_{t-i}]^T$ the partial likelihood function:

$$f_{\theta_i}(y_j | y_{j-1}, \ldots, y_1) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_j - x_{j,i}^T \phi_i)^2}{2\sigma^2} \right) = \exp\left(-\frac{1}{2} \log(2\pi\sigma^2) - \frac{(y_j - x_{j,i}^T \phi_i)^2}{2\sigma^2} \right)$$

$$f_{\theta_i}(y_j | y_{j-1}, \ldots, y_1) = \exp(h(\theta_i)_j)$$

The quantity $y_j - x_{j,i}^T \phi_i$ is called the residual for the j-th observation, measures the vertical distance between the data point y_j and the hyper plane $x_{j,i}^T \phi_i$, and thus assesses the degree of fit between the actual data and the model.

It follows, that the log-likelihood function to be maximized is

$$g(\theta_i) = \sum_{j=n+1}^{T} h(\theta_i)_j = -(T-n-1)\frac{1}{2}\log(2\pi\sigma^2) - \sum_{j=n+1}^{t} \frac{(y_j - x_{j,i}^T \phi_i)^2}{2\sigma^2}$$

Now let $X_i = [x_{T,i}, x_{T-1,i}, \ldots, x_{n+1,i}]^T$ be an $((T-n-1)\times(i+1))$ matrix and $y = [y_T, y_{T-1}, \ldots, y_{n+1}]^T$ be an $T-n-1$ length vector. With this

$$g(\theta_i) = -(T-n-1)\frac{1}{2}\log(\sigma_i^2) - \frac{1}{2\sigma^2}(y - X_i\phi_i)^T(y - X_i\phi_i)$$

The vector of $\widetilde{\phi_i}$ which maximizes $g(\theta_i)$ is called the maximum-likelihood estimator for ϕ_i.

By the chain rule

$$\frac{dg(\theta_i)}{d\phi_i} = -\frac{1}{2\sigma^2}\frac{d(y - X_i\phi_i)^T(y - X_i\phi_i)}{d(y - X_i\phi_i)}\frac{d(y - X_i\phi_i)}{d\phi_i} = -\frac{1}{2\sigma^2}2(y - X_i\phi_i)^T(-X_i) = \frac{1}{\sigma^2}(y - X_i\phi_i)^T X_i$$

If X_i is not degenerate, meaning it has rank i, then $\widetilde{\phi_i}$ is optimal if and only if

$$y - X_i\widetilde{\phi_i} = 0$$

$$y = X_i\widetilde{\phi_i} \Rightarrow X_i^T y = X_i^T X_i\widetilde{\phi_i} \Rightarrow (X_i^T X_i)^{-1} X_i^T y = \widetilde{\phi_i}$$

$\widetilde{\phi_i}$ indeed yields the global maxima, as the objective function is concave: $\frac{dg^2(\theta_i)}{d\phi_i^2} = -\frac{1}{\sigma^2}X^T X$, which is a negative semi definite matrix.

Regarding σ^2 one can state that

$$\frac{dg(\theta_i)}{d\sigma^2} = -(T-n-1)\frac{1}{2\sigma^2} + \frac{1}{2}\frac{1}{\sigma^4}(y - X_i\widetilde{\phi_i})^T(y - X_i\widetilde{\phi_i})$$

$\widetilde{\sigma_i^2}$ is optimal if and only if

$$-(T-n-1)\frac{1}{2\widetilde{\sigma_i^2}} + \frac{1}{\widetilde{\sigma_i^4}}(y - X_i\widetilde{\phi_i})^T(y - X_i\widetilde{\phi_i}) = 0$$

$$\widetilde{\sigma_i^2} = \frac{(y - X_i\widetilde{\phi_i})^T(y - X_i\widetilde{\phi_i})}{T-n-1}$$

The formula fulfills our intuition, as the variance maximum-likelihood estimator of σ^2, is the average of the squared sum of the residuals for the non-deterministic part of the time series.

The product $X_i^T X_i$ is symmetrical matrix, and its inverse is the *cofactor* matrix of ϕ_i[1], closely related to its covariance matrix, C_{ϕ_i}[2]. The matrix $(X_i^T X_i)^{-1} X_i^T$ is called the *Moore-Penrose pseudo inverse* matrix (Moore (1920)) of X_i[3]. From now on refer to this matrix as $X_{MP,i}^{-1}$. This formulation highlights the point that estimation can be carried out, if and only if, there is no perfect multicollinearity between the different lags of y (which would cause the normal matrix to has no inverse).

After we have estimated $\widetilde{\phi}_i$, the predicted values will be $\hat{y} = X_i \widetilde{\phi}_i = P_i y$, where $P_i = X_i (X_i^T X_i)^{-1} X_i^T = X_i X_{MP,i}^{-1}$ is the projection matrix onto the space V spanned by the columns of X_i. Thus the dimension of its target space is p. The matrix P_i is symmetric and idempotent ($P_i^2 = P_i$) and relate to the data matrix X_i via the identity $P_i X_i = X_i$.

The error terms can be reformulated as $y - P_i y = (I - P_i) y$. $I - P_i$ or the annihilator matrix is also a symmetric projector, as $(I - P_i)^2 = I - P_i$. The rank of the annihilator matrix is $T - n - 1 - i$. It also holds that $(I - P_i) X_i = 0$. Hence $\widetilde{\sigma_i^2} = \frac{(y - X_i \widetilde{\phi_i})^T (y - X_i \widetilde{\phi_i})}{T-n-1} = \frac{((I-P_i)y)^T ((I-P_i)y)}{T-n-1} = \frac{((I-P_i)(X_i\phi_i+\varepsilon))^T ((I-P_i)(X_i\phi_i+\varepsilon))}{T-n-1} = \frac{((I-P_i)\varepsilon)^T ((I-P_i)\varepsilon)}{T-n-1} = \frac{\varepsilon^T (I-P_i)^T (I-P_i)\varepsilon}{T-n-1} = \frac{\varepsilon^T (I-P_i)\varepsilon}{T-n-1}$.

3.4 Finite results

In the followings, I analyze the connection between $\widetilde{\theta}_i$ and θ. For that purpose, we have to distinguish three different cases, where $i < p$, $i = p$, and $i > p$. ε is a $(T - n - 1)$ length vector in this setup, where every entry is normally distributed with mean zeros, and the covariance matrix $\sum$ is $\sum = \text{diag}(\sigma^2)$.

First case: i equals to p

Note that if i equals to p, then

$$\widetilde{\phi}_p = X_{MP,p}^{-1} \cdot y = X_{MP,p}^{-1} (X_p \phi + \varepsilon) = \phi + X_{MP,p}^{-1} \varepsilon$$

[1] I will show some asymptotic results about the cofactor matrix later on.

[2] which will turn out to be $\widetilde{\sigma_i^2} (X_i^T X)_i^{-1}$

[3] The nomenclature is appropriate, as X_i is not square in almost all the cases, hence X_i is not invertible. However $X_{MP,i}^{-1} X_i = 1_i$

ε is normally distributed, so is its linear combination. X is a random matrix, however $\mathbb{E}X_{MP,p}^{-1}\varepsilon|X$ is the null vector, which yields that $\mathbb{E}X_{MP,p}^{-1}\varepsilon = 0$ via the tower rule of conditional expectation. As a result$\widetilde{\phi_p}$ is an unbiased estimator of the variable ϕ. The Gauss-Markov theorem states, that in this case the estimator is BLUE: best linear unbiased estimator (Theil (1971)).

Proposition 6. *$\widetilde{\phi_p}$ centered at ϕ and normed by its standard deviation is jointly t-distributed.*

Proof. $\widetilde{\phi_p} = X_{MP,p}^{-1} \cdot y$ where $y \sim \mathcal{N}(X_p\phi, \text{diag}(\sigma^2)_{T-n-1})$. As a result, $\widetilde{\phi_p} \sim \mathcal{N}(\phi, C)$, where

$$C = (X_p^T X_p)^{-1} X_p^T \sum X_p (X_p^T X_p)^{-1} = (X_p^T X_p)^{-1} X_p^T \sigma^2 1_{T-n-1} X (X_p^T X_p)^{-1} = \sigma^2 (X_p^T X_p)^{-1}$$

[4]Now let us move on to the variance estimator. $I - P_p$ is a projection as discussed, so the only eigenvalues it can have are 0 and 1. Hence there exists a orthogonal matrix V, such that $V^T(I-P_p)V = \Delta = \text{diag}(1,\dots,1,0,\dots,0)$. The trace of a projection matrix is the dimension of the target space, so $tr(I-P_p) = T-n-1-p$. Since the trace is also the sum of the eigenvalues multiplied by their multiplicity, we necessarily have that 1 is an eigenvalue with multiplicity $T-n-1-p$ and zero is an eigenvalue with multiplicity p. Thus Δ has $T-n-1-p$ ones in the diagonal.

Now let $K = V^T\hat{\varepsilon} = V^T(1-P_p)y = V^T(1-P_p)(X_p\phi_p + \varepsilon) = V^T(1-P_p)\varepsilon$. Since $(1-P)\varepsilon \sim \mathcal{N}(0,(1-P_p)\sigma^2 1_{T-n-1})$, we have $K \sim \mathcal{N}(0,\sigma^2\Delta)$[5], and therefore $K_{T-n-p} = K_{T-n-p+1} = K_T = 0$. It follows that $\frac{||K||_2^2}{\sigma^2} \sim \chi^2_{T-n-1-p}$. Further, as V is orthogonal $||K||^2 = ||\hat{\varepsilon}||^2$, thus

$$\frac{\widetilde{\sigma_p^2}}{\sigma^2} \cdot (T-n-1) = \frac{(y - X_p\widetilde{\phi_p})^T(y - X_p\widetilde{\phi_p})}{\sigma^2} \sim \chi^2_{T-n-1-p} \tag{3.4.1}$$

Of course it implies that $\mathbb{E}\widetilde{\sigma_p^2} = \sigma^2\frac{T-n-1-p}{T-n-1}$, so the maximum-likelihood estimator for σ^2 is biased (but asymptotically unbiased). It is also true that $Var(\widetilde{\sigma_p^2}) = 2\frac{T-n-1-p}{(T-n-1)^2}\sigma^2$.

$X_p^T X$ is a positive semi-definite, symmetrical matrix, so there exists a representation $R^{-1}DR = X_p^T X_p$, where $D = \text{diag}(d_l)_{l=1,\dots,p}$ is diagonal. Define $\sqrt{(X_p^T X_p)^{-1}}$ as $R^{-1}DR$.

From the conditional multi-normality of $\widetilde{\phi_p}$, it follows is that $\frac{1}{\sigma}\sqrt{(X_p^T X_p)^{-1}}(\widetilde{\phi_p} - \phi_p) \sim \mathcal{N}(0,1_p)$. Additionaly, from 3.4.1 it turns out that $\frac{1}{\widetilde{\sigma_p}}\sqrt{(X_p^T X_p)^{-1}}(\widetilde{\phi_p} - \phi_p) \sim t_{T-n-1-p}$.

[4]$\sum$is the covariance matrix of ε, not the summing operator
[5]$\sigma^2 V^T(1-P_p)^T 1_{T-n-1}(1-P_p)V = \sigma^2 V^T(1-P_p)V = \sigma^2\Delta$

Hence the claim. Note that the covariance matrix of $\frac{1}{\widetilde{\sigma_p}}\sqrt{(X_p^T X_p)^{-1}}\widetilde{\phi_p}$ is given by diag $\left(\frac{T-n-3-p}{T-n-1-p}\right)$.[6]

□

Later on, it is more interesting if we investigate the properties of $\xi_p = \frac{1}{\widetilde{\sigma_p}}(\widetilde{\phi_p} - \phi_p)$. As a linear combination of the previous random vector, its expectation is still zero, and its covariance matrix is diag $\left(\frac{T-n-3-p}{T-n-1-p}\right) X_p^T X_p$.

Second case: i is less than p

In the case when i is less than p, X_i is a $(T-n-1 \times (i+1))$ matrix, but the "real" matrix $X_p$would have a shape $(T-n-1 \times (p+1))$. Observe that the first i columns in X_i and X_p are identical. Denote the "leftover" $((T-n-1) \times (p-i))$ matrix with A. Construct X_p as $X_i \cdot [I_i, B]$, where I_i is the $i \times i$ identity matrix. By easy calculations, $B = (X_i^T X_i)^{-1} X_i^T A = X_{MP,i}^{-1} A$, therefore

$$\widetilde{\phi_i} = X_{MP,i}^{-1}(X_i \cdot [I_i, X_{MP,i}^{-1} A] \cdot \phi + \varepsilon) = [I_i, X_{MP,i}^{-1} A] \cdot \phi + X_{MP,i}^{-1} \varepsilon$$

This of course yields that $\widetilde{\phi_i}$ is biased unless all $(\phi_j)_{j>i} = 0$, as for example $\widetilde{\phi_1} = \phi_1 + \alpha\phi_3 + \beta\phi_4 + \gamma\phi_5 + c\varepsilon_1$, with some constant α, β, γ coming from the first row of $X_{MP}^{-1} A$.

Proposition 7. *$\widetilde{\phi_i}$ centered at its expectation and normed by its standard deviation is jointly t-distributed.*

Proof. From Proposition 6 we have a unitary matrix W, such that $W^T(I-P_i)W = \Delta$, where Δ is a diagonal matrix filled with ones and zeros. The trace of a projection matrix is the dimension of the target space, so $tr(I-P_i) = T-n-1-i$. Thus Δ has $T-n-1-i$ ones in the diagonal.

Now let $K = W^T \hat{\varepsilon} = W^T(I-P_i)y \overset{(I-P_i)X_i=0}{=} W^T(I-P_i)(X_i \cdot [I_i, X_{MP,i}^{-1} A] \cdot \phi + \varepsilon) = W^T(I-P_i)\varepsilon$. Since $(I-P_i)\varepsilon \sim \mathcal{N}(0, (I-P_i)\sigma^2 1_{T-n-1})$, we have $K \sim \mathcal{N}(0, \sigma^2\Delta)$, and therefore $K_{T-n-i} = K_{T-n-i+1} = K_T = 0$. It follows that $\frac{||K||_2^2}{\sigma^2} \sim \chi^2_{T-n-1-i}$. Further, as W is orthogonal $||K||^2 = ||\hat{\varepsilon}||^2$, thus

$$\frac{\widetilde{\sigma_i^2}}{\sigma^2} \cdot (T-n-1) = \frac{(y - X_i\widetilde{\phi_i})^T (y - X_i\widetilde{\phi_i})}{\sigma^2} \sim \chi^2_{T-n-1-i} \tag{3.4.2}$$

[6] All the results are conditioned on X

Of course it implies that $\mathbb{E}\widetilde{\sigma_i^2} = \sigma^2 \frac{T-n-1-i}{T-n-1} \geq \mathbb{E}\widetilde{\sigma_p^2}$, so the maximum-likelihood estimator for σ^2 is biased. Also, $Var(\widetilde{\sigma_i^2}) = 2\frac{T-n-1-i}{(T-n-1)^2}\sigma^2 \geq Var(\widetilde{\sigma_p^2})$.

From this point the proof continues accordingly. □

Based on claim 7 , we can tell that $\frac{1}{\widetilde{\sigma_i}}\sqrt{(X_i^T X_i)^{-1}}(\widetilde{\phi_i} - [1_i, X_{MP,i}^{-1}A]\cdot\phi_p) \sim t_{T-n-1-i}$. Note that the actual covariance matrix of $\frac{1}{\widetilde{\sigma}}\sqrt{(X_i^T X_i)^{-1}}\widetilde{\phi_i}$ is $diag(\frac{T-n-3-i}{T-n-1-i})$.

As previously, if $\xi_i = \frac{1}{\widetilde{\sigma}}(\widetilde{\phi_i} - [I_i, X_{MP,i}^{-1}A]\cdot\phi_p)$, then $\mathbb{E}\xi_i = 0$ and $Var(\xi_i) = \text{diag}\left(\frac{T-n-3-i}{T-n-1-i}\right) X_i^T X_i$.

Third case: i is greater than p

In the last case, we investigate what happens when i is actually greater than p. In this case $X_p = X_i \cdot \begin{pmatrix} 1_p \\ 0 \end{pmatrix}$, so

$$\widetilde{\phi_i} = X_{MP,i}^{-1}(X_p = X_i \cdot \begin{pmatrix} 1_p \\ 0 \end{pmatrix} \cdot \phi_p + \varepsilon) = X_p = \begin{pmatrix} 1_p \\ 0 \end{pmatrix}\phi + X_{MP,i}^{-1}\varepsilon = \begin{matrix} \phi \\ 0 \end{matrix} + X_{MP,i}^{-1}\varepsilon,$$

so $\widetilde{\phi_i}$ is an unbiased estimator of the variable ϕ, if we extend $\phi$with $p-i$ zeros.

Proposition 8. *$\widetilde{\phi_i}$ centered at its expectation and normed by its standard deviation is jointly t-distributed.*

Proof. From 6 we have a orthogonal matrix Z, such that $Z^T(I-P_i)Z = \Delta$, where Δ is a diagonal matrix filled with ones and zeros. The trace of a projection matrix is the dimension of the target space, so $tr(I-P_i) = T-n-1-i$. Thus Δ has $T-n-1-i$ ones in the diagonal.

Now let $K = Z^T\hat{\varepsilon} = Z^T(I-P_i)y \overset{(I-P_i)X_i=0}{=} Z^T(I-P_i)(X_i \cdot \begin{pmatrix} 1_p \\ 0 \end{pmatrix} \cdot \phi + \varepsilon) = W^T(I-P_i)\varepsilon$. Since $(I-P_i)\varepsilon \sim \mathcal{N}(0, (I-P_i)\sigma^2 1_{T-n-1})$, we have $K \sim \mathcal{N}(0, \sigma^2\Delta)$, and therefore $K_{T-n-i} = K_{T-n-i+1} = K_T = 0$. It follows that $\frac{||K||_2^2}{\sigma^2} \sim \chi^2_{T-n-1-i}$. Further, as W is orthogonal $||K||^2 = ||\hat{\varepsilon}||^2$, thus

$$\frac{\widetilde{\sigma_i^2}}{\sigma^2}\cdot(T-n-1) = \frac{(y - X_i\widetilde{\phi_i})^T(y - X_i\widetilde{\phi_i})}{\sigma^2} \sim \chi^2_{T-n-1-i} \tag{3.4.3}$$

Of course it implies that $\mathbb{E}\widetilde{\sigma_i^2} = \sigma^2\frac{T-n-1-i}{T-n-1} \leq \mathbb{E}\widetilde{\sigma_p^2}$, so the maximum-likelihood estimator

for σ^2 is biased. Also, $Var(\widetilde{\sigma_i^2}) = 2\frac{T-n-1-i}{(T-n-1)^2}\sigma^2 \leq Var(\widetilde{\sigma_p^2})$.

From this point the proof continues accordingly. □

According to aforementioned proposition, $\frac{1}{\widetilde{\sigma_i}}\sqrt{(X_i^T X_i)^{-1}}(\widetilde{\phi_i} - \begin{pmatrix} 1_p \\ 0 \end{pmatrix} \cdot \phi) \sim t_{T-n-1-i}$. The covariance matrix of $\frac{1}{\widetilde{\sigma_i}}\sqrt{(X_i^T X_i)^{-1}}\widetilde{\phi_i}$ is diag $\left(\frac{T-n-3-i}{T-n-1-i}\right)$. If $\xi_i = \frac{1}{\widetilde{\sigma_i}}(\widetilde{\phi_i} - \begin{pmatrix} 1_p \\ 0 \end{pmatrix} \cdot \phi_p)$, then $\mathbb{E}\xi_i = 0$ and $Var(\xi_i) = \text{diag}\left(\frac{T-n-3-i}{T-n-1-i}\right) X_i^T X_i$.

3.4.1 Conclusion

To sum up the findings related to finite sampling, see Table 3.1.

Table 3.1: Maximum-likelihood estimators in terms of order i

	$i < p$	$i = p$	$i > p$
$\widetilde{\sigma_i^2}$			
$\mathbb{E}$	Asy. unbiased	Asy. unbiased	Asy. unbiased
	-	$\geq$	$\geq$
Var	Asy. goes to 0	Asy. goes to 0	Asy. goes to 0
	-	$\geq$	$\geq$
$\widetilde{\phi_i}$			
$\mathbb{E}$	Finitely/Asy. biased	Unbiased	Unbiased
Var	$\leq$	$\leq$	-

Example. Table 3.2 outputs a single realization of the maximum likelihood estimators for a trajectory of a chosen $AR(4)$ process.

Table 3.2: maximum likelihood estimators for an $AR(4)$ model in terms of order i

This table shows a single realization of the maximum likelihood estimators for all the parameters of an autoregressive process, in terms of its order. In the experiment, I assumed that Y follows a 4-order autoregressive trend, with parameters $\alpha = 5, \phi_1 = 0.8, \phi_2 = 0.25, \phi_3 = -0.4, \phi_4 = 0.1, \sigma^2 = 1.8$. The rows stands for the modelled order, and the columns contain the estimated parameters. I simulated a 10000 length $AR(4)$ trajectory, with an initial value of $y_0 = 2$.

i	$\widetilde{\alpha}$	$\widetilde{\phi_1}$	$\widetilde{\phi_2}$	$\widetilde{\phi_3}$	$\widetilde{\phi_4}$	$\widetilde{\phi_5}$	$\widetilde{\phi_6}$	$\widetilde{\phi_7}$	$\widetilde{\sigma^2}$
1	4.389	0.781	-	-	-	-	-	-	2.0490
2	4.244	0.755	0.033	-	-	-	-	-	2.0469
3	5.631	0.766	0.280	-0.327	-	-	-	-	1.8289
4	5.083	0.798	0.252	-0.401	0.097	-	-	-	1.8117
5	5.090	0.798	0.252	-0.401	0.099	-0.001	-	-	1.8119
6	5.105	0.798	0.252	-0.402	0.099	0.001	-0.003	-	1.8122
7	5.100	0.798	0.252	-0.402	0.100	0.001	-0.004	0.001	1.8124

3.5 Asymptotic results

3.5.1 The cofactor matrix

If we assume, that y is already in its stationary state, and the process runs since infinite amount of time, then we can make the following conclusions. Let $m = \frac{\alpha}{1-\sum_{i=1}^{p}\phi_i}$ denote the stationer expectation of y. Let γ_m define the covariance structure of y in terms of the lag $m = 0, \ldots, \infty$ (γ_0 is the stationary variance D^2). Of course, $(\gamma_m)_{m>p} = 0$.

As described above, for the fixed order $i \in \mathscr{P}$, $X_i = \begin{pmatrix} 1 & y_{T-1} & \ldots & y_{T-i} \\ 1 & y_{T-2} & \ldots & y_{T-i-1} \\ 1 & y_{T-3} & \ldots & y_{T-i-2} \\ \ldots & \ldots & \ldots & \ldots \end{pmatrix}$ with $T-n-1$ rows and $i+1$ columns. Let $f : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$, such that $f(X)_{k,l} = \frac{X_{k,l}}{\sqrt{m}}$. Now treat T as if it was a feature of the matrix X_i. Accordingly, denote the matrix by $X_{T,i}$. For every fixed $T \in \mathbb{N}$, the followings hold for the matrix $N = f(X_{T,i})^T f(X_{T,i})$:

- $N_{1,1} = \frac{1}{T-n-1}$

- Whenever $k>1, l>1,\ N_{k,l}=N_{l,k}=\begin{cases}\frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}^2 & if\ k=l\\ \frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}y_{t-l} & otherwise\end{cases}$

- If $k=1$ or $l=1$, $N_{k,l}=N_{l,k}=\frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}$

By stationarity, $\mathbb{E}\lim_{T\to\infty}\frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}^2=\lim_{T\to\infty}\mathbb{E}\frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}^2=\lim_{T\to\infty}(\mathbb{D}^2y_T+\mathbb{E}^2y_T)=\lim_{T\to\infty}\mathbb{D}^2y_T+\lim_{T\to\infty}\mathbb{E}^2y_T=\gamma_0+m^2=D^2+(\frac{\alpha}{1-\sum_{i=1}^{p}\phi_i})^2$. The lim and the expectation is interchangeable in the first round, as e.g $\frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}^2\le \max_{t\in(0,T)}(y_t^2)$, and by stationarity $\max_{t\in(0,T)}(y_t^2)$ has finite expectation, so the dominated convergence theorem is applicable. Also, by the strong low of large numbers, $\frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}^2$ converges to its expectation strongly.

By the very same reason, $\frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}y_{t-l}$ converges almost surely to $\gamma_{|k-l|}+m^2$, while $\frac{1}{T-n-1}\sum_{t=n+1}^{T} y_{t-k}$ converges to $m=\frac{\alpha}{1-\sum_{i=1}^{p}\phi_i}$ with probability one.

As a result, we can state that

$$\lim_{T\to\infty} N=\lim_{T\to\infty} f(X_{T,i})^T f(X_{T,i})=\lim_{T\to\infty}\frac{1}{T-n-1}X_{T,i}^T X=\begin{pmatrix}1 & m & m & m & \dots\\ m & \gamma_0+m^2 & \gamma_1+m^2 & \gamma_2+m^2 & \dots\\ m & \gamma_1+m^2 & \gamma_0+m^2 & \gamma_1+m^2 & \dots\\ m & \gamma_2+m^2 & \gamma_1+m^2 & \gamma_0+m^2 & \dots\\ \dots & \dots & \dots & \dots & \dots\end{pmatrix}. \tag{3.5.1}$$

The matrix has $i+1$ columns and rows. Hence the cofactor matrix $(X_{T,i}^T X)^{-1}$ indeed does reflect the covariance structure of y. Observe, that the entries do not depend on the particular choice of i!

3.5.2 Parameter estimations

We have seen in proposition 7, 6 and 8, that the theoretical distribution of $\widetilde{\phi_i}$ is t, regardless of what $i\in\mathscr{P}$ is. Trivially, they obtain different degrees of freedom, but these are monotonically increasing in T.

Lemma. *A t-distributed random variable converges weakly to a standard normal, as the degree goes to infinity.*

Proof. Let $(X_n)_{n\in\mathbb{N}}$ have standard t-distribution , so it can be formulated as $X_n = \frac{Z}{\sqrt{\frac{\chi^2_n}{n}}}$, where $Z \sim \mathcal{N}(0,1)$, and χ^2_n is a Chi-square distributed random variable with n degrees of freedom (independent of Z). By definition $\chi^2_n = \sum_1^n N_i^2$, where $N_i \sim i.i.d\,\mathcal{N}(0,1)$. When n tends to infinity χ^2_n converges in probability to its expectation 1 (weak law of large numbers). As a consequence, by Slutsky's theorem, X_n converges weakly to Z. □

Based on Lemma 3.5.2:

$$\forall i \in \mathscr{P}, \lim_{T\to\infty} \frac{1}{\widetilde{\sigma}_i}(\widetilde{\phi}_i - \mathbb{E}\widetilde{\phi}_i) \sim \mathcal{N}(0, X_i^T X_i),$$

or according the 3.5.1 result, and introducing *RSS*, the residual sum of squares ($RSS = (y - X_i\widetilde{\phi}_i)^T(y - X_i\widetilde{\phi}_i) = \widetilde{\sigma}_i^2 \cdot (T-n-1)$), this yields

$$\forall i \in \mathscr{P}, \lim_{T\to\infty} \frac{1}{\sqrt{RSS}}(\widetilde{\phi}_i) \sim \mathcal{N}(\mathbb{E}\widetilde{\phi}_i, \begin{pmatrix} 1 & m & m & \dots \\ m & \gamma_0 + m^2 & \gamma_1 + m^2 & \dots \\ m & \gamma_1 + m^2 & \gamma_0 + m^2 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix})$$

3.6 L_2 norm loss - asymptotic partial flatness

Assume we have a complete history of a single trajectory y of $Y \sim AR(p)$ until $T \in \mathbb{N}$, and we would like to predict the next value of y for $T+1$. Assume Y is determined by $\theta = \theta_p$. y_{T+1} is given by $x^T_{T+1,p}\phi_p + \varepsilon$, where $\varepsilon \sim \mathcal{N}(0,\sigma^2)$. On the other hand, the forecast based on our model is ,$\hat{y}_i = x^T_{T+1,i}\widetilde{\phi}_i$.

In order to determine which hyper-parameter i to choose as best, we must investigate how the respective predictions behave at future unknown values. An equivalent formulation is that knowing how the loss functions corresponding to the different orders are distributed across the orders gives us all information we need to distinguish among the hyper-parameters.

According to the model's assumption, the loss function associated to i is given by by $L_i = (y_{T+1} - \hat{y}_i)^2$. The difference $y - \hat{y}$ is asymptotically normally distributed (as a linear combination of normal variables). Thus $y - \hat{y} \approx \mathbb{E}(y-\hat{y}) + \mathcal{N}_i(0, Var(y-\hat{y})) = \mathbb{E}(y-\hat{y}) + std(y -$

$\hat{y})\mathcal{N}_i(0,1)$. It means that the loss associated with the modeling order i is $L_i \approx \mathbb{E}^2(y-\hat{y}_i) + Var(y-\hat{y}_i)\chi_i^2 + 2\mathbb{E}(y-\hat{y}_i)\cdot std(y-\hat{y}_i)\mathcal{N}_i(0,1)$. The different χ_i^2 and the normal variables are dependent, hence the notation. The expectation of such a loss is $\mathbb{E}L_i \approx \mathbb{E}^2(y-\hat{y}_i) + Var(y-\hat{y})$ approximately.

For all $i \geq p, i \in \mathscr{P}$, $\mathbb{E}^2(y-\hat{y}_i) = 0$, as the estimators are unbiased. For the other hyper-parameters in the initial set of $\mathscr{P}$, the square of the expectation of the prediction residual is greater than zero. Asymptotically, the variance of the predicted underlying shrinks to zero (as the variance of the estimators also), thus

$$\forall i \in \mathscr{P},\ \mathbb{E}L_i = \begin{cases} a^2, & if\ i < p \\ 0 & otherwise \end{cases}.$$

Corollary. *The theoretical P^* set for an autoregressive process is asymptotically the hyper-parameter set $A = i \geq p, i \in \mathscr{P}$. In the long run, the Model Confidence Set is A at any significance level.*

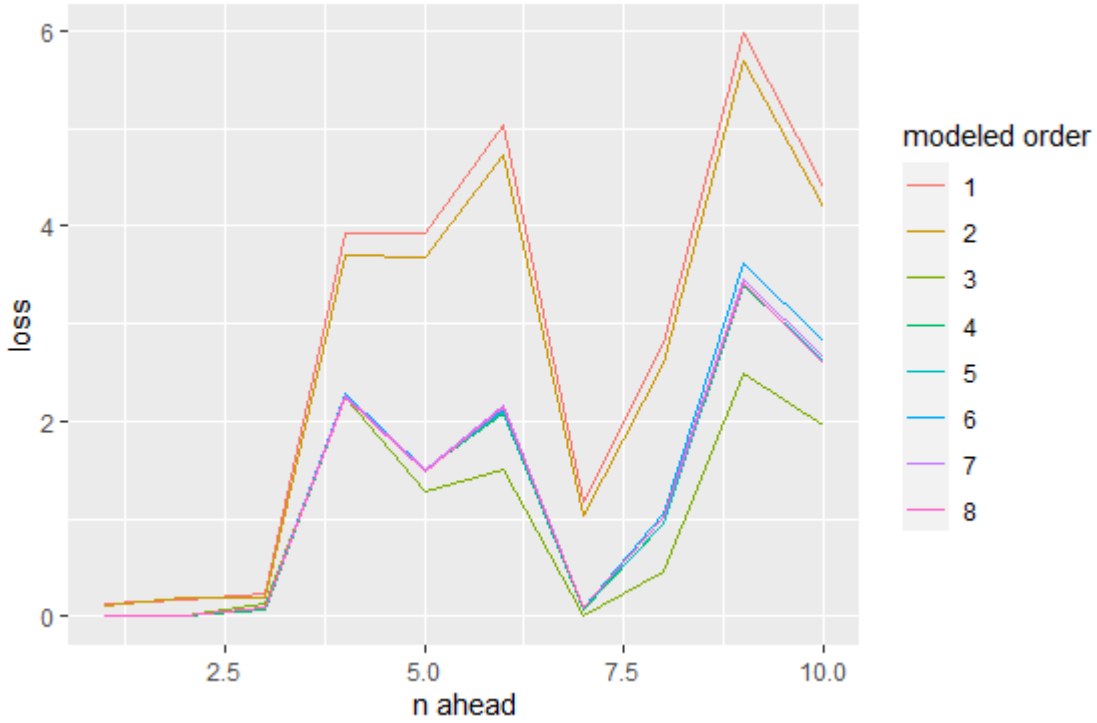
Also, for the P^* set we can tell that $\forall i \in P^*, L_i \approx \sigma^2\chi_i^2$, as the variance of the new observation is σ^2 and the variance of the prediction shrinks to zero asymptotically. Note that for this set χ_i^2 is highly dependent across the different orders (correlation is nearly one). The loss associated with the MCS superior set is $L_{MCS} \approx \frac{1}{|\mathscr{P}|}\sum_{i=1}^{|\mathscr{P}|} L_i \approx \sigma^2\frac{1}{n}\sum_{1}^{n}\chi_i^2$. Hence $\mathbb{E}L_{MCS} = \mathbb{E}L_i = \sigma^2$ for all $i \in P^*$.

3.7 A simulation result

Figure 3.7.1 displays a realization of loss values in a 10 ahead prediction setup. I simulated 3000 levels of a trajectory of an AR process with $(\alpha, \phi_1, \phi_2, \phi_3, \phi_4, \sigma^2) = (5, 0.8, 0.25, -0.4, 0.1, 1.2)$. 10 observations have been taken out as the validation set, and on the remaining 2990 observations, I applied the maximum likelihood estimators for the parameters, and predicted the remaining 10 level via them. The L_2-norm loss of the predictions are displayed in the figure. In this particular trajectory the model generated by the hyper-parameter (order) 3 yielded the minimal mean loss. As we can see, the losses are indeed highly correlated.

Figure 3.7.2 shows how the MCS and grid search losses are distributed in a simple experiment with 10.000 observations. In each round I simulated 1000 levels of a trajectory of an AR process with $(\alpha, \phi_1, \phi_2, \phi_3, \phi_4, \sigma^2) = (5, 0.8, 0.25, -0.4, 0.1, 1.2)$. The last observation has

Figure 3.7.1: A realization of loss values associated with the different orders in a 10-ahead prediction setup



The interactive version of the plot can be found by clicking here.

been taken out as the test set. 10 observations have been taken out as the validation set, and on the remaining 989 observations, I applied the maximum likelihood estimators for the parameters, and predicted the remaining 10 level via them. Grid search chooses the parameter that yielded the minimal mean loss, while MCS contains the losses associated with the orders 4 to 8. I predicted the last observation via the grid search and the MCS model.

The mean MCS loss in this particular experiment is 3.0513. The respective mean grid search loss is 3.0883. The t-statistics that tests whether the means of the losses equal has a p-value less than 0.0001. Thus we can reject the hypothesis with almost 100% confidence that the MCS loss equals to the grid search loss.

Out of the 10 000 observations, 5258 times the MCS algorithm yielded the less loss. If the grid search and the MCS approach would uniformly generate the less loss, than the number of such a difference count would follow a binomial distribution with parameters $n = 10000$ and $p = 0.5$. Accordingly, the corresponding 5258 number would have been sampled with probability less than 0.000001. Thus we can reject the statement at almost 100% confidence that MCS and grid search are equally explanatory with respect to the future predictions. Figure 3.7.3 confirms these findings. It is seen, that in this particular simulation grid search

produced higher losses more times than MCS.

Figure 3.7.2: Joint density function of the MCS and grid search loss based on a simulation

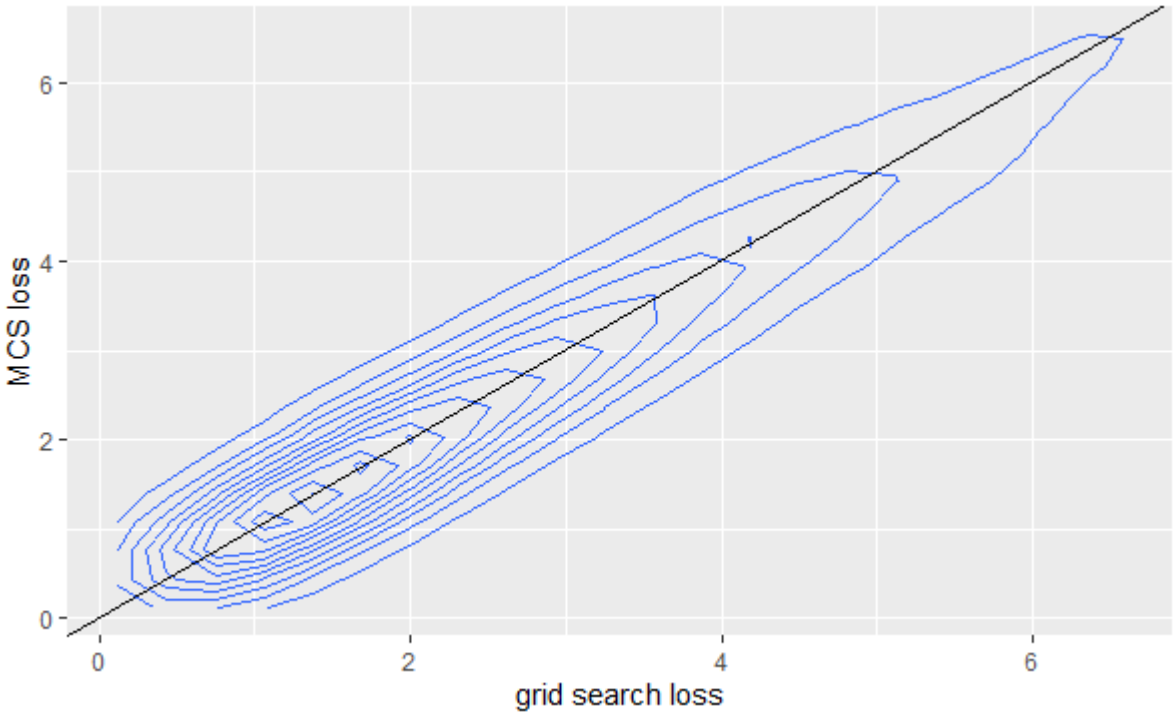


Figure 3.7.3: Relative count for given loss levels for MCS and grid search losses based on a simulation

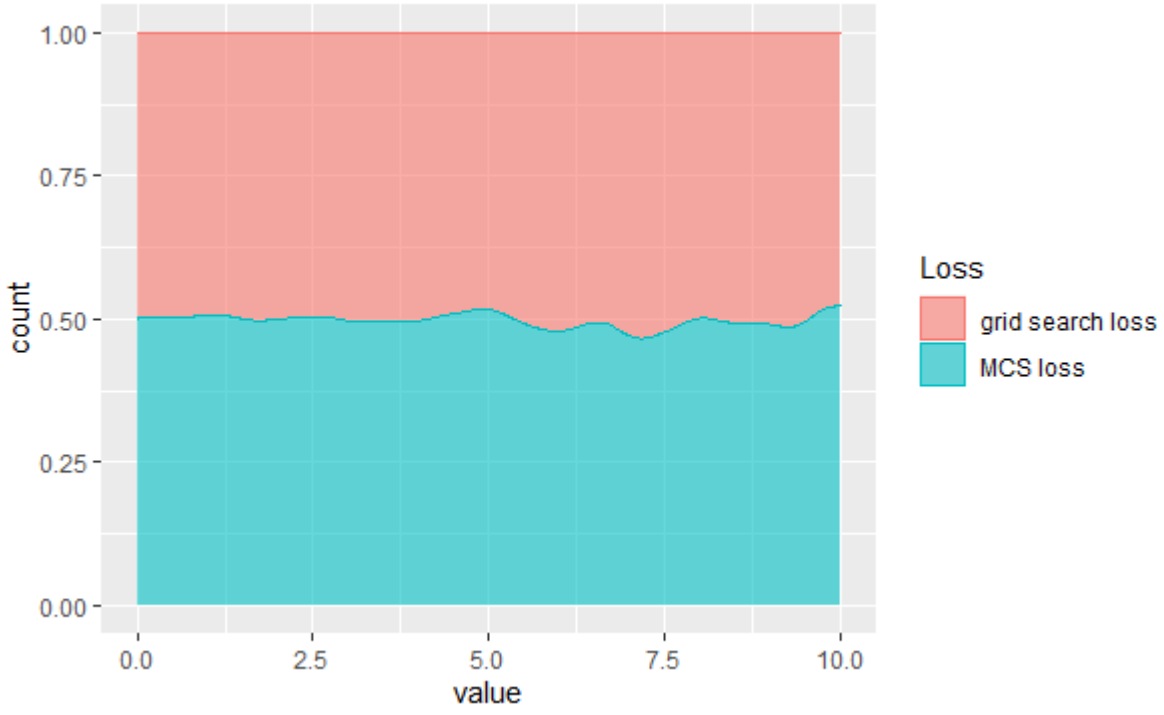
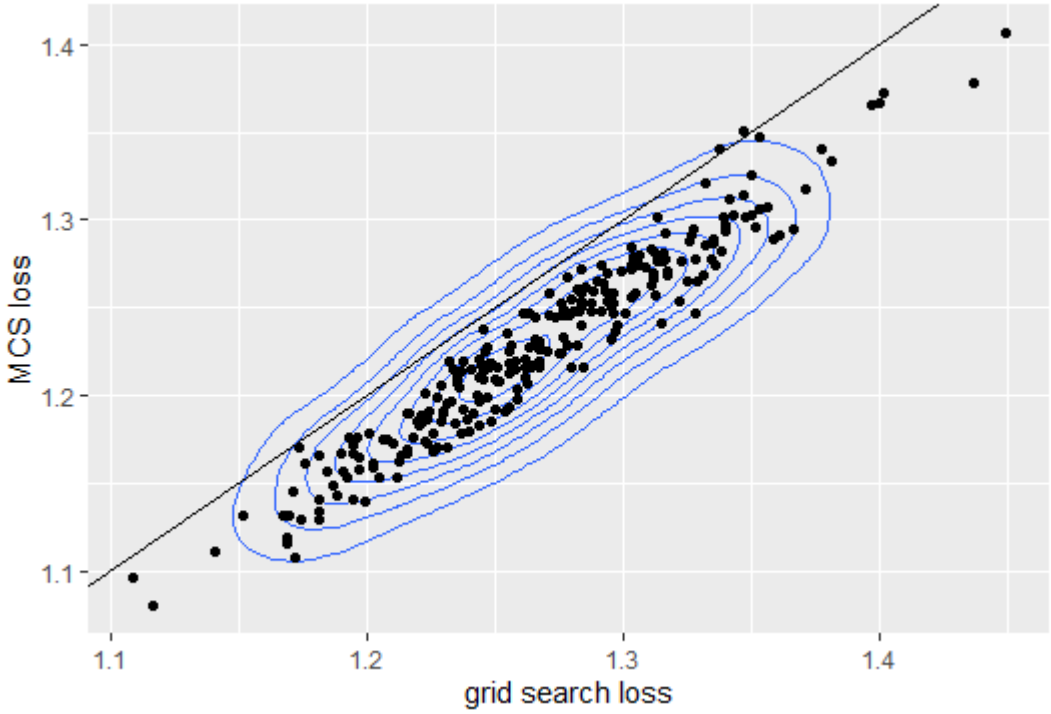


Figure 3.7.4 is the main conclusion of the theoretical experiment. I applied the same experiment as before but instead of once, 250 times. The figure shows how the mean MCS and the mean grid search losses are distributed. One point in the below plot represents an 1000 length loop error vector.

The MCS loss was 1.22 in average, which is almost $\mathbb{E}\sigma^2\chi_i^2$, as discussed above. The difference from 1.2 is statistically not significant, as the t-statistic that tests whether the loss is 1.2 has a p-value 0.25.

It turned out that the two mean losses are significantly differ from each other at a 1% significance level. The p-value associated to the t-statistics that tests whether the loss difference equals to zero is 0.008. Based on the previous findings, this concludes that MCS algorithm yields significantly less loss than grid search.

Figure 3.7.4: Mean MCS and grid search losses based on a simulation



An interactive version of this plot is seen by clicking *here*.

Chapter 4

Empirical investigation - in light of predicting infectious diseases

At the time of writing this paper (early spring of 2020), there is a considerable outbreak of the novel corona virus disease (Covid-19). Hence the incentive is given to test whether MCS can beat the grid search method in forecasting the epidemic curve.

4.1 The MN-SEIR model extended to capture contact tracing and isolation

4.1.1 Introducing the SEIR model family

The most commonly used framework for epidemiological systems is the susceptible-infectious-recovered (SIR) class of models, in which the host population is categorized according to infection status as either susceptible, infectious or recovered, respectively (Kermack & McKendrick (1927)). Subsequent refinements of the model have incorporated an additional exposed (infected but not yet infectious) class so that the model became SEIR.

One of the fundamental mathematical assumptions in such models is that the rate of leaving the exposed and the infectious class is constant, irrespective of the period already spent in that class. While this makes the mathematics very easy to handle, this assumption gives rise to exponentially distributed latent and infectious periods, which is epidemiologically unrealistic for most infections (Sartwell *et al.* (1950), Bailey (1975), Wearing *et al.* (2005a)). A more

realistic approach is to assume that the probability of leaving a class is a function of time spent inside, which is small at first and increasing after the mean infectious/latent period is reached. For example in the case of the infectious class one could model such a phenomena with the following integral equation:

$$I(t) = I(0)P_I(t) + \int_{\tau=0}^{t} \frac{\beta_I S(\tau) I(\tau)}{N} P_I(t-\tau) d\tau$$

where $I(t)$ is the number of infectious people at time t, $P_I(t)$ is the probability of remaining infectious after τ periods, β_I is the infection transmission rate parameter of the model to estimate, $S(t)$ is the number of susceptible people at time t, and N is the number of the population. As a result, the expected time being infectious denoted with $\frac{1}{\gamma}$ is $\int_0^\infty P_I(\tau) d\tau$ (Hethcote & Tudor (1980)). Note that the time being infectious is non-negative, hence $\mathbb{E}t_{infect.} = \int_0^\infty 1 - F_{t_{infect}}(\tau) d\tau$, so the probability density function of this time is given by $p_I(t) = -\frac{dP_I(\tau)}{d\tau}$.

In terms of the exposed class $E(t)$, the expected time spent in it is often denoted by $\frac{1}{\sigma}$. A similar differential equation holds there, of course applying a different multiplier β_E, and a different probability density function $p_E(t)$.

From modeling purposes, the question is how to choose $p_I(t)$ and $p_E(t)$ to match the empirical evidences.

4.1.2 Configure the MN-SEIR modeling setup

In the following paragraph I demonstrate the setup for the infectious class, however the exact same holds for the exposed one as well, but with different parameters of course.

Lloyd (2001) propose that a realistic or empirically provable distribution can be obtained by choosing $p(t)$ to be a gamma probability density function, with parameters γ and n (σ and m for the exposed class). The expectation of such a variable is $\frac{1}{\gamma}(\frac{1}{\sigma})$, which fulfills our requirements described in the previous subsection. However, using the fact that the gamma distribution is a sum of exponentially distributed variables we introduce n (m) different infectious (exposed) class $(I^{(1)}, \ldots, I^{(n)})$ $((E^{(1)}, \ldots, E^{(m)}))$, with $n\gamma$ ($m\sigma$) being the rate of sequential progression through the sub classes. Equivalently, the time spent in each infectious class is exponentially distributed with $\frac{1}{n\gamma}$ $(\frac{1}{m\sigma})$ expectation, so that the whole infectious period is $\frac{1}{\gamma}$

($\frac{1}{\sigma}$) in expectation, and it follows the proposed gamma distribution. The effects of n on the distribution of the infectious period is seen at figure 4.1.1. Table 4.1 summarizes the estimates of the hyper-parameters m, n for some famous diseases.

Table 4.1: Latent and infectious period duration together with the respective gamma parameters

Estimates of expected time spent in the exposed and the infectious class, respectively, together with the associated gamma-distribution parameter for four diseases. The latent period and infectious period columns are measure in days.

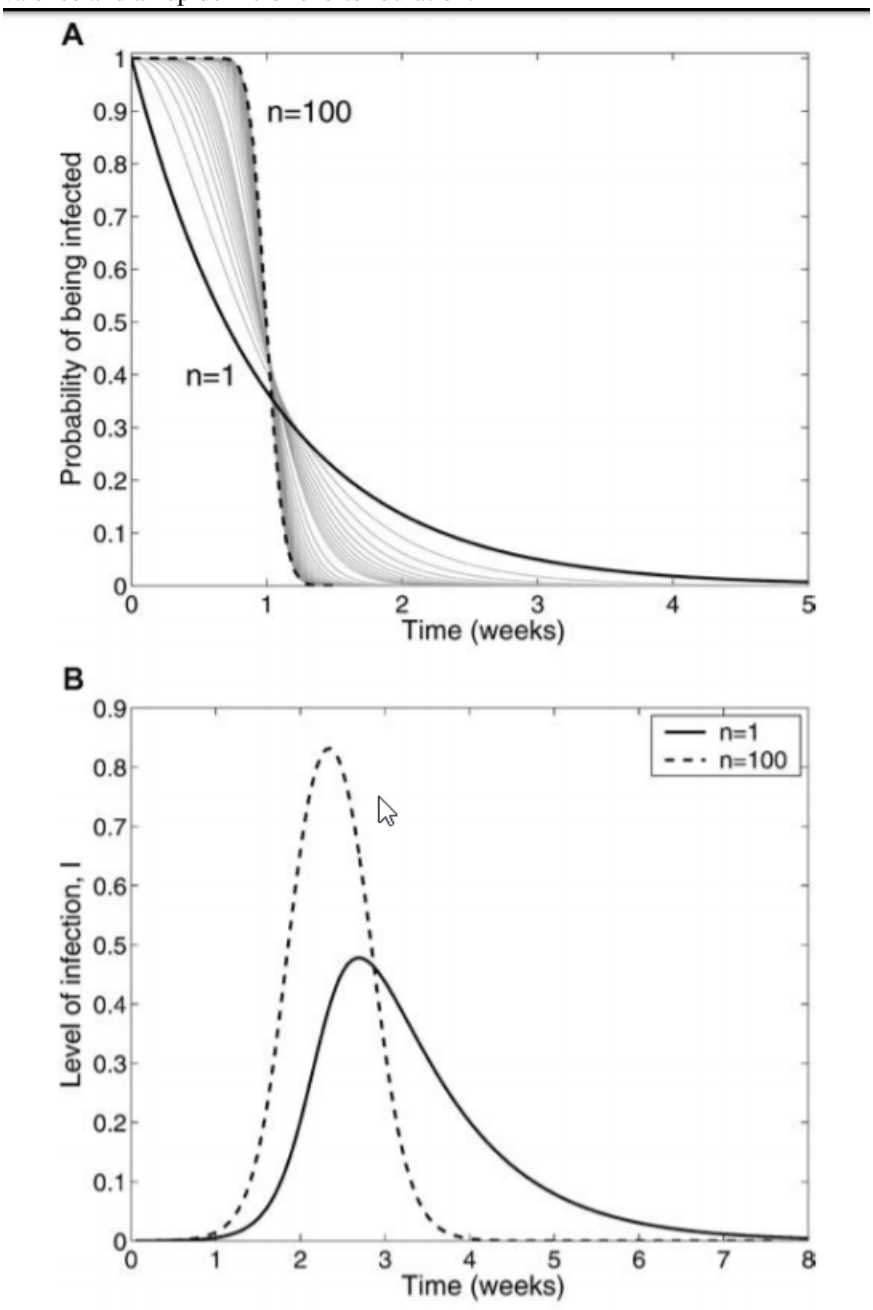
Disease	Latent period $\frac{1}{\sigma}$	m	Infectious period $\frac{1}{\gamma}$	n
Measles	8	~20	5	~20
Foot-and-Mouth	3.5	13	4.3	17
SARS	5.36	2	5-6	3
Smallpox	14	40	8.6	4

Source: Wearing *et al.* (2005a), 3rd page

Figure 4.1.1: Gamma-distributed infectious periods and their effects on the epidemic curve

In figure *A* the change in probability of remaining infectious is seen as a function of time when the number of subdivisions within the infected class increases from $n = 1$ to $n = 100$. Note that the mean duration of being infectious is the same for all n: 1 week. If $n = 1$ the distribution of this time is exponential, and as n grows, the time shrinks to constant one week.

In figure *B*, we can see the consequences of changes in n for the SIR-type epidemic. For the same basic reproductive ratio $R_0 = 5$, and the same average infectious period $\gamma = 1$, larger values of n lead to a steeper increase in prevalence and an epidemic of shorter duration.



Source: Wearing *et al.* (2005a), 3rd page.

4.1.2.1 Contact tracing and quarantine

To incorporate contact tracing and isolation into the model, Wearing *et al.* (2005b) proposed another extension of the model, while still dealing with gamma-distributed latent and infectious periods. In this model, isolation of newly infectious cases occurs at a daily rate d_I after a delay of τ_D days, which represents a period when infected individuals are infectious but asymptomatic or undetectable (I_A).

A fraction of q of those who had contact with an infectious and symptomatic person (I_S) (but did not contract the infection) are removed to the quarantined susceptible class. An identical fraction of newly exposed individuals are also quarantined.

4.1.2.2 Why this model?

One can surely ask why I have chosen such a model to forecast the Covid-19 related issues. The answer is two folded.

First, it is based on logical assumptions about how an epidemic evolves. This means that configuring the right parameters enables us to better understand the underlying disease. This is undoubtedly an advantage compared to the machine learning algorithms which are generally considered as "black-box" learning algorithms.

Secondly, we lack the data. For an arbitrary machine learning algorithm to learn we need a great amount of data, however we only have at most 2, 2.5 months daily observations in the Covid-19 disease. This makes all the mentioned learning algorithms fail in practice.

4.1.3 Main modeling equations

The following equations were introduced by Wearing *et al.* (2005a), however I made a slight change in the formulation.

To catch that people does not contact as often as before, as the epidemic increases, I assumed that the contact number decreases to a minimum of k_0. The minimal contact size is meaningful, because for example people has to do some shopping even in the middle of an epidemic. The rate how $k(t)$ shrinks is λ.

$$\frac{dk}{dt} = -\frac{k(t)-k_0}{\lambda}$$

$$\frac{dS}{dt} = -\frac{(k(t)bI(t)+qk(t)(1-b)I_S(t))S(t)}{N} + \frac{qk(t)(1-b)S(t-\tau_Q)I_S(t-\tau_Q)}{N}$$

$$\frac{dS_Q}{dt} = \frac{qk(t)(1-b)S(t)I_S(t)}{N} - \frac{qk(t)(1-b)S(t-\tau_Q)I_S(t-\tau_Q)}{N}$$

$$\frac{dE_1}{dt} = \frac{k(t)b(I(t)-qI_S(t))S(t)}{N} - m\sigma E_1(t)$$

$$\frac{dE_i}{dt} = m\sigma E_{i-1}(t) - m\sigma E_i(t), \quad i = 2,...,m$$

$$\frac{dI_{A,1}}{dt} = m\sigma E_m(t) - n\gamma I_{A,1}(t) - P_{I,1}(t)$$

$$\frac{dI_{A,i}}{dt} = n\gamma I_{A,i-1}(t) - n\gamma I_{A,i}(t) - P_{I,i}(t), \quad i = 2,...,n$$

$$\frac{dI_{S,1}}{dt} = P_{I,1}(t) - (n\gamma + d_I)I_{S,1}(t)$$

$$\frac{dI_{S,i}}{dt} = P_{I,i}(t) + n\gamma I_{S,i-1}(t) - (n\gamma + d_I)I_{S,i}(t), \quad i = 2,...,n$$

$$\frac{dQ}{dt} = \frac{qk(t)bS(t)I_S(t)}{N} + d_I I_S(t)$$

$$\frac{dR}{dt} = n\gamma(I_{A,n}(t) + I_{S,n}(t))$$

where

$$P_{I,i}(t) = m\sigma E_m(t-\tau_D)e^{-n\gamma\tau_D}\frac{(n\gamma\tau_D)^{i-1}}{(i-1)!}$$

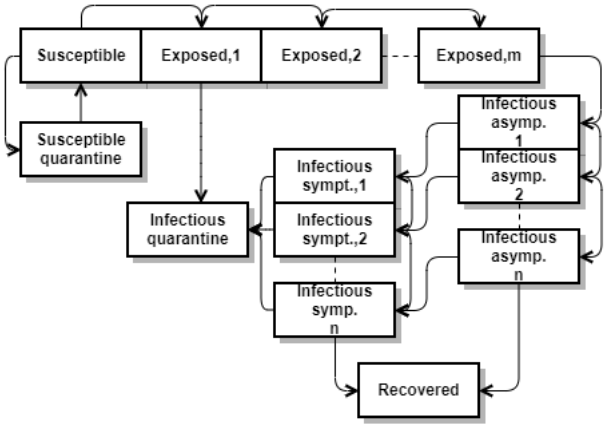
is the expected number of infectious individuals at time t that are still in infectious class i after a fixed delay of τ_D days.

Of course, $I_A(t) = \sum_{i=1}^{n} I_{A,i}(t)$, $I_S(t) = \sum_{i=1}^{n} I_{S,i}(t)$ and $I(t) = I_A(t) + I_S(t)$. Observe that here R represents those that recovered or died before they could be isolated/quarantined. Hence the total number of "recovered"[1] will be $Q+R$ as the epidemic dies out, since Q keeps track of all those infected who are isolated, and effectively removed from the infectious population. N is the total population size. Figure 4.1.2 describes the different states and the possible direction of the flow among them.

To conclude, observe that the parameters and the of the above described system of differential equations are $K_0, \lambda, b, q, \sigma, \gamma, \tau_D, \tau_Q, d_I, N$. From this list, N and τ_Q can be handled as constants: N is the population size and τ_Q is the susceptible quarantine period which is fixed by the government. The hyper-parameters of the system are m and n.

[1] I consider dead as recovered, as the only meening of it mathematically is they left the population without remaining susceptible or infectious.

Figure 4.1.2: The MN-SEIR model extended to match contact tracing and isolation



Own figure, implemented according to Wearing *et al.* (2005b)

4.1.4 Limitations of the model

The presented MN-SEIR model extended to capture contact tracing and isolation is a great mathematical model, but has some limitations. Always keep these limitations in mind when analyzing the results.

First and foremost, mathematically speaking the model is the solution of the above defined system of differential equation, hence a vector of functions. These functions are deterministic, thus knowing all the parameters of them is equivalent to knowing the whole past, present and future states of the different classes. As the model does not allow randomness, we view it as the description of the expected system dynamics.

Secondly, the SIR family of models assume homogeneous mixing across the different population groups. It says that an individual is for example as likely to meet an elderly as a young man, irrespective of their attributes. This is obviously not true in real life. Again, all we can say that it captures the system dynamics in expectation. Actually, there are models that differentiate within the population groups, e.g. the K-SEIR model presented by Lipton & de Prado (2020), but they are extremely hard and computationally exhaustive to estimate.

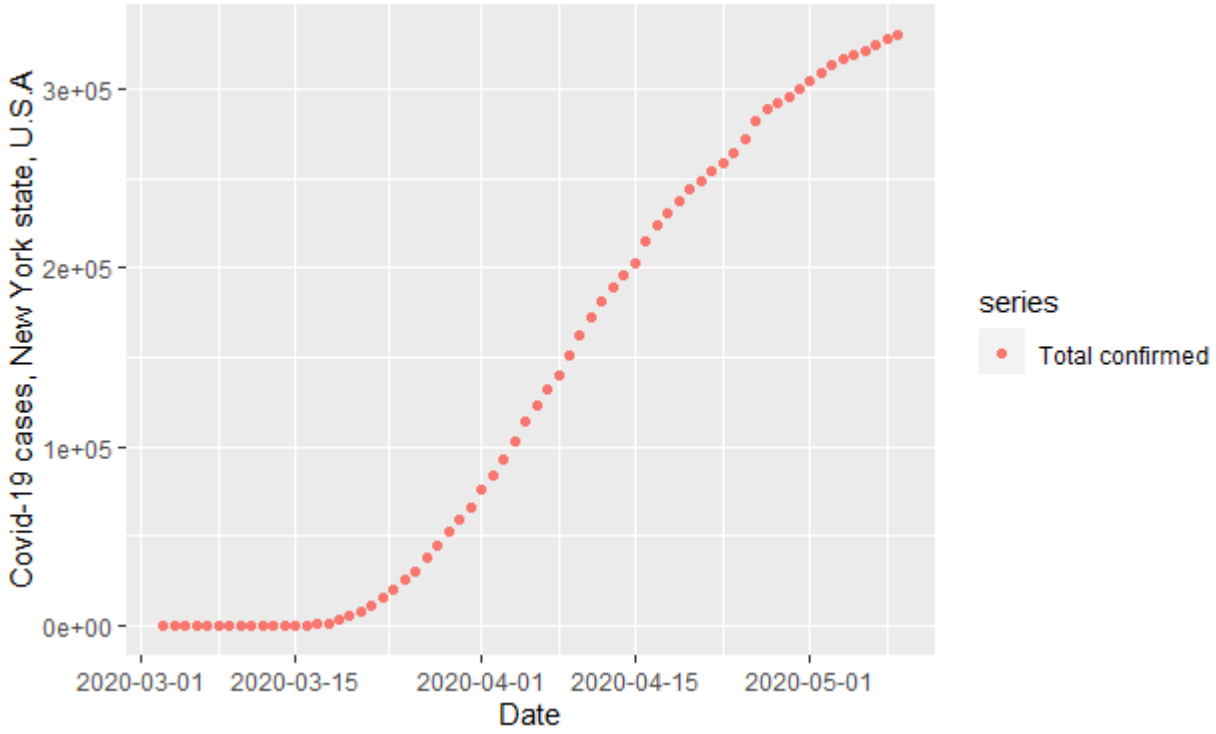
Lastly, I assumed that the number of contacts in time is an exponentially decreasing function, which is true for the first part of an epidemic, but loses its truth afterwards. Hence the model is not really applicable to forecast very far in the future. Instead, it is more than great to make statements about the upcoming several days.

4.2 Data

I made the empirical investigation on the data that describes how New York state, U.S.A has been affected by the Covid-19 corona virus. The data is provided by the John Hopkings Whiting School of Engineering, Center of Systems Science and Engineering. They reserve a public GitHub repository (click here to see), where an update of the numbers happen on a daily basis.

I downloaded the global time series data from the repository on the 9th of May, 2020 and filtered out all the entries that were based on non New York state numbers. The filtered data set is seen at the online appendix of this paper (click here), under the name *curve.csv.* The data covers all the officially confirmed (infectious, recovered plus dead) individuals on each day. Based on the data, the first observation of a Covid-19 infectious individual happened on the third of March, 2020. On the 9th of May, 2020 the confirmed cases summed up to 330.407. See Figure 4.2.1 that displays the data.

Figure 4.2.1: Covid-19 total confirmed cases in New York state, U.S.A as of the 9th of May, 2020



An interactive version of the plot can be found *here*.

4.3 The research

I implemented the above mentioned dynamic system in *R,* using the *deSolve* package[2]. The package provides the *dede* function, which is useful for solving delayed differential equations, like these. The code is found the online appendix linked to this paper.

As noted in the introduction, the hyper-parameter optimization involves an inner search where all the parameters of the problem gets configured for a particular grid point m, n. For this optimization problem I applied the *optimParallel* function of the *optimParallel* package[3] to utilize the computational power of the chosen computer.

The data set contains 68 days of observations. The training set X^{train} is the first 44 observations. On this set, I perform the inner optimization to optimize the parameters within each group m,n. The next 15 observations are in the validation set of the model, where predictions of each models and actual values are evaluated with respect to the L_2 norm loss function.

The predictions on the validation set make it possible to apply the grid search and the Model Confidence Set. Grid search chooses the model, that minimizes the sum of the validation errors, while MCS returns a set of models that contains the best at a significance level α. I have chosen α to be 10%.

I compare the effectiveness of the two hyper-parameter optimization procedure by predicting levels for the remaining 9 observations: grid search uses the superior model to predict, while MCS uses all the ones in the superior set and then takes an average.

4.4 Results

4.4.1 Test errors - the competitor models

The best performed model (on the validation set) was the one with hyper-parameters $m = 3, n = 4$[4]. Its mean squared error on the validation set is 8.000.831 which is an approximate 2.829 absolute difference in forecasting the total cases. The average case number on the validation set was ~260.119, thus the error is approximately 1%.

[2]https://cran.r-project.org/web/packages/deSolve/index.html

[3]https://cran.r-project.org/web/packages/optimParallel/index.html

[4]In the appendix, you can see some explanatory figures related to the best performed model to understand some aspects of the Covid-19 disease.

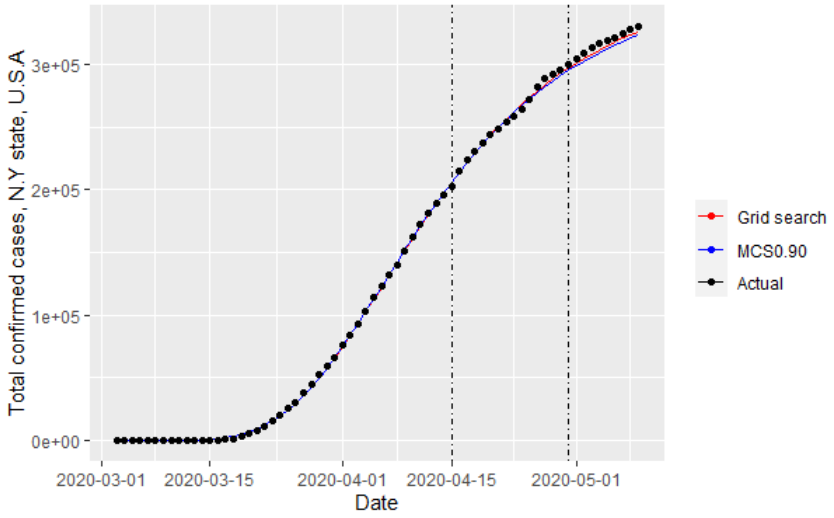
The Model Confidence set contains 8 models at a 10% significance level. These 8 are $\begin{pmatrix} m \\ n \end{pmatrix} = \left(\begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 3 \\ 2 \end{pmatrix}, \begin{pmatrix} 3 \\ 3 \end{pmatrix}, \begin{pmatrix} 3 \\ 4 \end{pmatrix}, \begin{pmatrix} 3 \\ 5 \end{pmatrix}, \begin{pmatrix} 5 \\ 2 \end{pmatrix}, \begin{pmatrix} 5 \\ 3 \end{pmatrix}, \begin{pmatrix} 5 \\ 5 \end{pmatrix} \right)$. The set produces a mean squared error of 12.173.559, which is an approximate 3489 absolute difference in forecasting the total cases. The result is not surprising, as the minimum (which is chosen by grid search) is less than all the other errors. The MCS validation error is approximately 1.34%.

4.4.2 Validation errors - the final results

In the followings I present the forecast errors produced by the grid search and the MCS forecast. I have chosen two ways to look at the results, and in both ways I compared the two error vectors with a t-statistics. That is, the null-hypothesis was that difference of the error terms are zero, and the alternative hypothesis was its complement. The validation dataset contains 9 observations, thus the t-distribution has 8 degrees of freedom.

The MCS results are a slight worse than the grid search results with respect to the actual values of the time series. Grid search has produced a mean squared error of 156.152.219, while for MCS this number is 398.902.923. The difference is significant at any normal significance level. These levels are an approximate absolute differences of 3.9% and 6.3%, respectively. Figure 4.4.1 displays the forecasts.

Figure 4.4.1: Forecast Covid-19 cases New York state, U.S.A - MCS vs grid search
Total confirmed cases and their forecasts via grid search and MCS based on the MN-SEIR model. The two vertical lines separate the train, test and validation data sets. Grid search performed significantly better at any normal significance level.



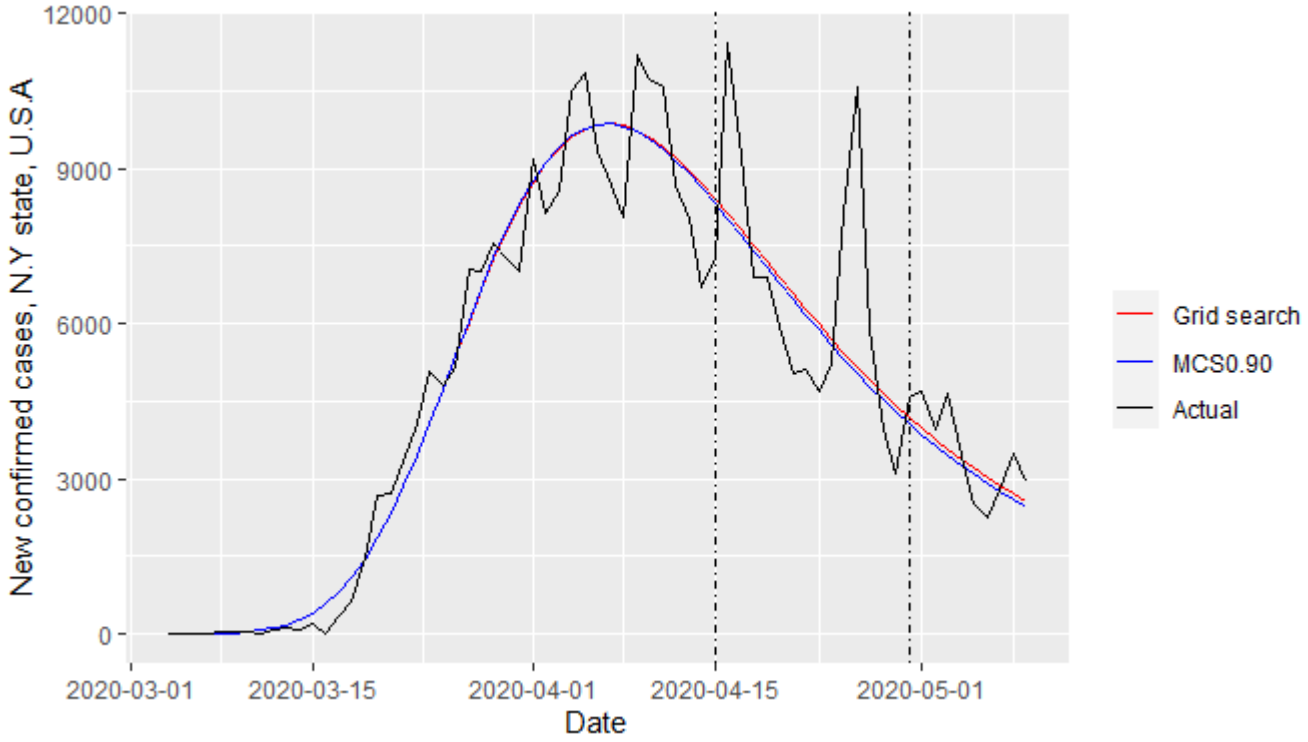
An interactive plot of the same figure can be accessed by clicking *here.*

However, by the nature of the problem, if a big shock happens, then that method wins, that was closer to the shock when it happened. This is equivalent to saying that the methods proposed are extremely sensitive to outliers. In our case, a shock happened in the validation set, which made the grid search won in forecasting the actual case number. All in all, we cannot rely on the cumulative confirmed cases when deciding which model is the better.

On the other hand, it is more interesting to investigate which method forecasts the new cases greater (i.e. the derivative of the function). This approach decreases the effect of an underlier. If we choose this approach, then MCS and grid search performs statistically indistinguishable at any normal significance level, as the p-value of the difference is 34.1%. The first produced an approximate 19.6% error, while the latter produced 18.3%. This approach is displayed at Figure 4.4.2.

Figure 4.4.2: Forecasting new confirmed Covid-19 cases, New York state, U.S.A

New confirmed cases and their forecasts via grid search and MCS based on the MN-SEIR model. The two vertical lines separate the train, test and validation data sets. The two approaches are statistically indistinguishable (p-value is 34.1%) with respect to the test errors.



See the interactive version of the figure by clicking *here*.

Chapter 5

Conclusion

This paper proposes using the novel Model Confidence Set approach to solve hyper-parameter optimization problems. The algorithm reacts to the hidden issues in the most frequently applied grid search method in the topic. That is, the latter approach does not take statistical confidence and finite sampling into consideration when choosing the minimum loss yielding hyper-parameter.

On the contrary, the MCS algorithm returns a set that contains the ground true hyper-parameter(s) at a given confidence level. Given a well-behaving δ-test and a coherent elimination rule, the set where the algorithm terminates scales with the confidence level and the sample size, while remains powerful. Within this superior set, the hyper-parameters are statistically indistinguishable at the given confidence level with respect to the out-of-sample user defined loss their models produce. Thus MCS is an extension of the grid search method taking the actual variance of the respective losses into consideration.

After mathematically describing how the proposed algorithm works, I have shed light on the application of the MCS along an encouraging example. If the underlying process we aim to forecast is an autoregression, then asymptotically the out-of-sample loss curve is partially flat, meaning that the best hyper-parameter set contains all the orders greater than or equal to the original order. In that case, MCS significantly outperforms its ancestor, as the minimum yielding hyper-parameter set is observed to be singleton only because of randomness.

The second part is an empirical investigation about the performance of MCS applied to predicting future levels of an infectious disease. At the time of writing this thesis, there is a significant outbreak of the novel Covid-19 epidemic, hence the incentive is given to investigate that. I have extended the famous SEIR-model to capture gamma distributed latent and

infectious periods as well as contact tracing and isolation. The model is then a solution of a complex dynamic system with 10 parameters and 2 hyper-parameters. I applied the optimization on official data describing how the disease evolved in New York state, U.S.A up to May 9, 2020. It turned out that the model has a great explanatory power. However, there are some limitations of the model, which are needed to be considered when analyzing the results.

In the validation dataset, we cannot reject the hypothesis that the L_2-norm loss produced by MCS and grid search are statistically different at any significance level. This can be explained by the lack of data: I only have approximately 2 months' daily observations as of writing this paper to analyze, which makes it hard to produce significant results.

Asymptotically, MCS shrinks to the superior hyper-parameter set with probability one. If that set is a singleton, then asymptotically MCS contains the exact same hyper-parameter as the one chosen by grid search. Hence the differences come out when a finite sample is considered. If that happens, MCS takes randomness into consideration, which makes it more robust than the traditional grid search. If the referred set is not a singleton, then MCS always outperforms the latter, as derived in the respective section of this paper.

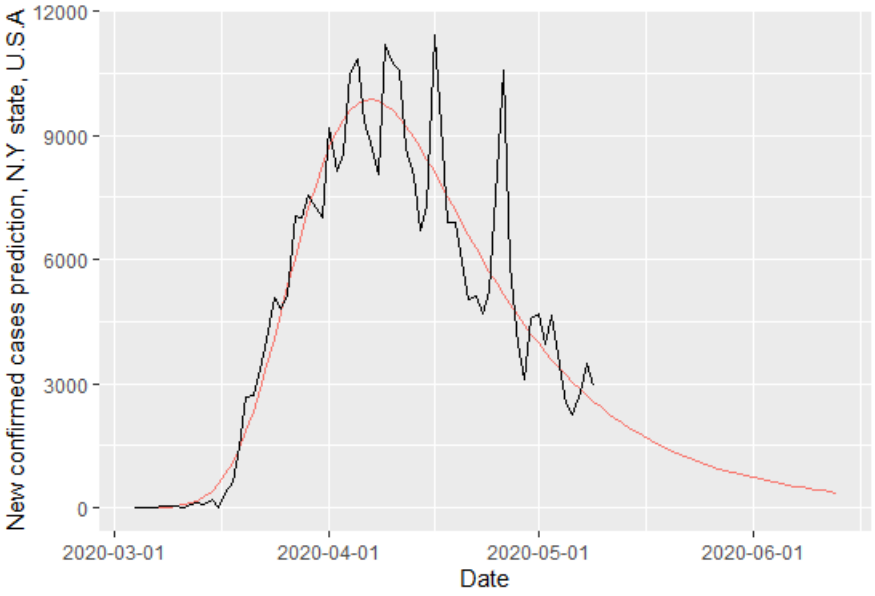
Chapter 6

Appendix

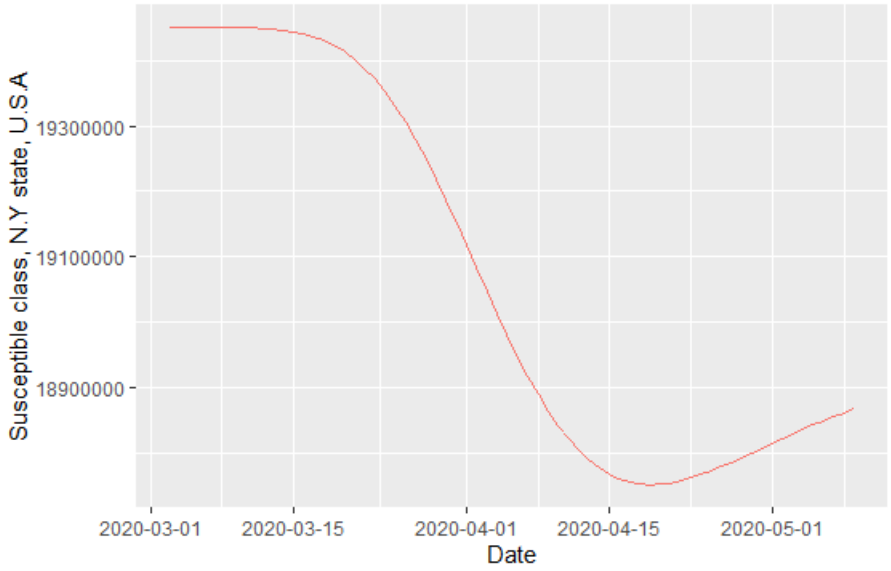
All the below figures represent the best fitting MN-SEIR model extended to catch contact tracing and isolation. They describe how each state evolved in the Covid-19 epidemic in New York state, U.S.A as of May 9, 2020 according to the presented dynamic system.

The best fitting one has three exposed and four infectious classes, and the parameters it has are the followings: $K_0 \approx 9.917$, $\lambda \approx 10.022$, $b \approx 0.0682$, $q \approx 0.3618$, $\sigma \approx 0.885$, $\gamma \approx 0.058$, $\tau_D \approx 0.6336$, $d_I \approx 0.8997$.

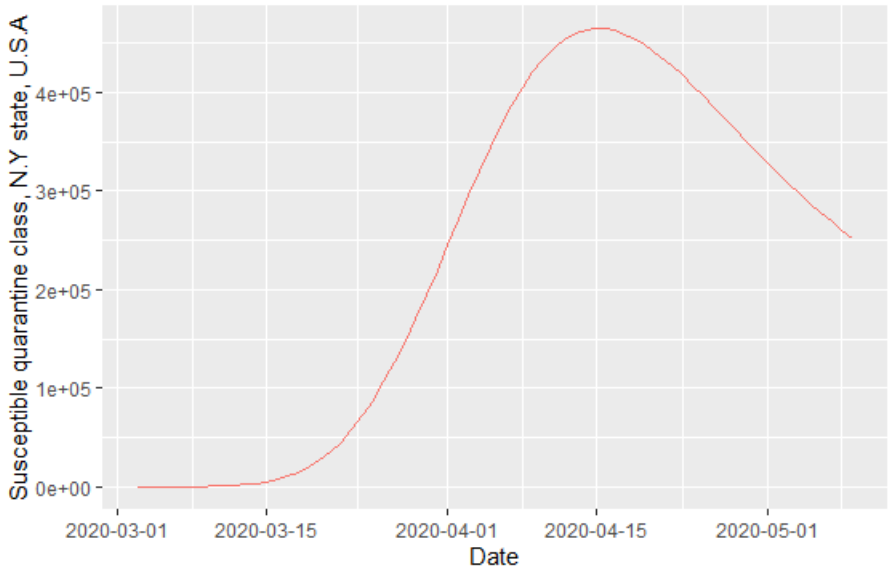
All of the plots listed below are available at *https://rpubs.com/kujbika.*



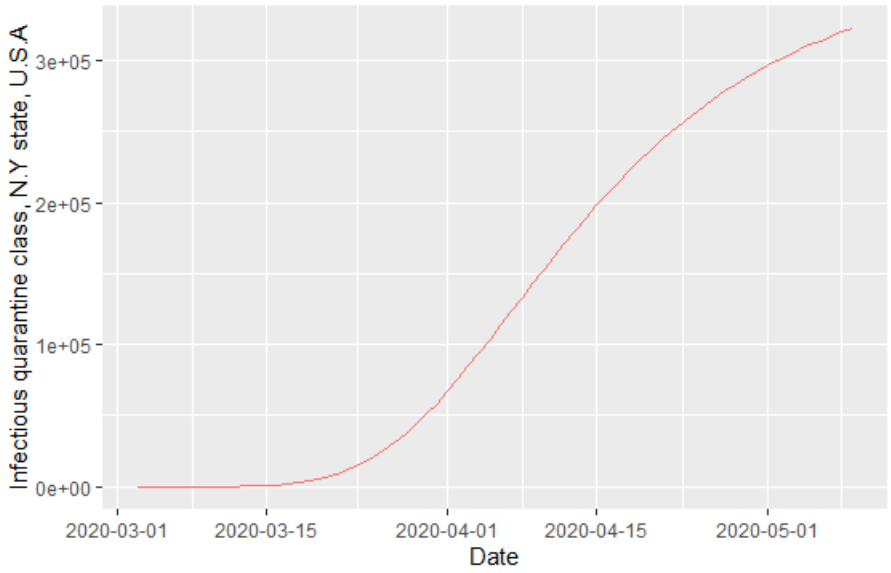
Prediction of new confirmed cases. An interactive version of this plot is seen by clicking *here*.



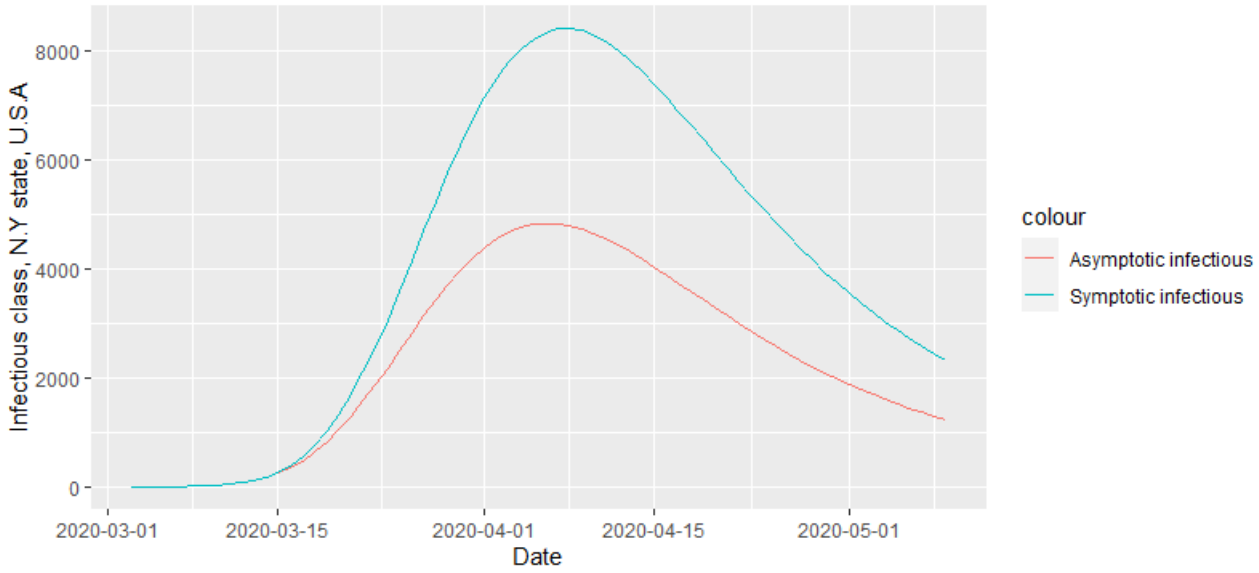
An estimation of the evolution of the susceptible class. An interactive version of this plot is seen by clicking *here*.



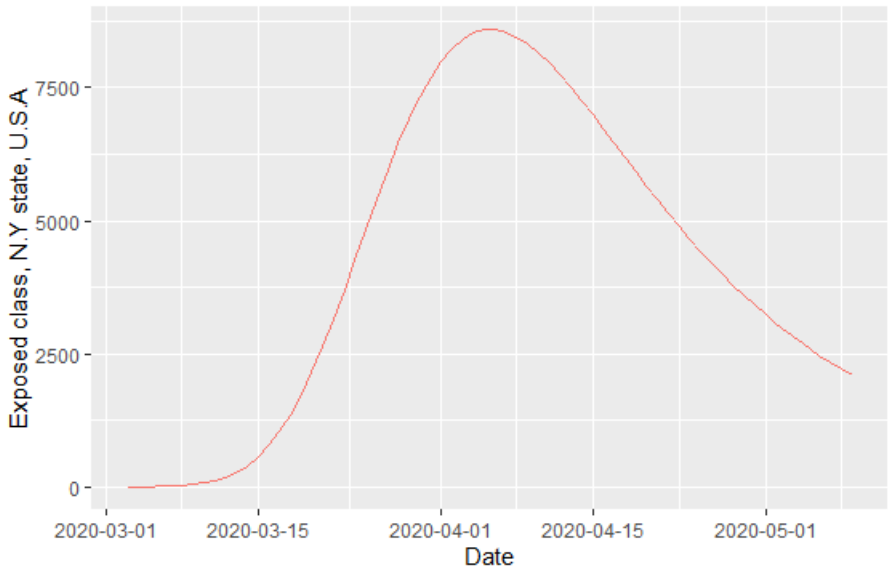
An estimation of the evolution of the susceptible quarantine class. An interactive version of this plot is seen by clicking *here.*



An estimation of the evolution of the infectious quarantine class. An interactive version of this plot is seen by clicking *here*.



As estimation of the evolution of the infectious classes. An interactive version of this plot is seen by clicking here.



As estimation of the evolution of the exposed class. An interactive version of this plot is seen by clicking here.

Bibliography

APARICIO, DIEGO, & LÓPEZ DE PRADO, MARCOS. 2018. How hard is it to pick the right model? MCS and backtest overfitting. *Algorithmic Finance*, **7**(1-2), 53–61.

BAILEY, NJJ. 1975. The Mathematical Theory ofInfectious Diseases 2nd edn (London: Griffin).

BELLMAN, RICHARD E. 1961. *Adaptive Control Processes: A Guided Tour*. Princeton university press, New Jersey.

BERGSTRA, JAMES, & BENGIO, YOSHUA. 2012. Random search for hyper-parameter optimization. *Journal of machine learning research*, **13**(Feb), 281–305.

BERNARDI, MAURO, & CATANIA, LEOPOLDO. 2018. The model confidence set package for R. *International Journal of Computational Economics and Econometrics*, **8**(2), 144–158.

DIEBOLD, FRANCIS X, & MARIANO, RS. 1995. Comparing forecast accuracy. *Journal of Business and*.

HANSEN, PETER R, LUNDE, ASGER, & NASON, JAMES M. 2011. The model confidence set. *Econometrica*, **79**(2), 453–497.

HETHCOTE, HERBERT W, & TUDOR, DAVID W. 1980. Integral equation models for endemic infectious diseases. *Journal of mathematical biology*, **9**(1), 37–47.

HINTON, GEOFFREY E, OSINDERO, SIMON, & TEH, YEE-WHYE. 2006. A fast learning algorithm for deep belief nets. *Neural computation*, **18**(7), 1527–1554.

HUTTER, FRANK, HOOS, HOLGER H, & LEYTON-BROWN, KEVIN. 2011. Sequential model-based optimization for general algorithm configuration. *Pages 507–523 of: International conference on learning and intelligent optimization*. Springer.

IOANNIDIS, JOHN PA. 2005. Why most published research findings are false. *PLos med*, **2**(8), e124.

KERMACK, WILLIAM OGILVY, & MCKENDRICK, ANDERSON G. 1927. A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, **115**(772), 700–721.

KIRKPATRICK, SCOTT, GELATT, C DANIEL, & VECCHI, MARIO P. 1983. Optimization by simulated annealing. *science*, **220**(4598), 671–680.

LAROCHELLE, HUGO, ERHAN, DUMITRU, COURVILLE, AARON, BERGSTRA, JAMES, & BENGIO, YOSHUA. 2007. An empirical evaluation of deep architectures on problems with many factors of variation. *Pages 473–480 of: Proceedings of the 24th international conference on Machine learning*.

LEEB, HANNES, & PÖTSCHER, BENEDIKT M. 2003. The finite-sample distribution of post-model-selection estimators and uniform versus nonuniform approximations. *Econometric Theory*, **19**(1), 100–142.

LIPTON, ALEX, & DE PRADO, MARCOS LÓPEZ. 2020. Exit Strategies for COVID-19: An Application of the K-SEIR Model (Presentation Slides). *SSRN Electronic Journal*, Apr.

LLOYD, ALUN L. 2001. Destabilization of epidemic models with the inclusion of realistic distributions of infectious periods. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, **268**(1470), 985–993.

MILLER, JAMES W. 1995. EXACT MAXIMUM LIKELIHOOD ESTIMATION IN AUTOREGRESSIVE PROCESSES. *Journal of Time Series Analysis*, **16**(6), 607–615.

MOORE, ELIAKIM H. 1920. On the reciprocal of the general algebraic matrix. *Bull. Am. Math. Soc.*, **26**, 394–395.

NAREYEK, ALEXANDER. 2003. Choosing search heuristics by non-stationary reinforcement learning. *Pages 523–544 of: Metaheuristics: Computer decision-making*. Springer.

NELDER, JOHN A, & MEAD, ROGER. 1965. A simplex method for function minimization. *The computer journal*, **7**(4), 308–313.

POWELL, MICHAEL JD. 1994. A direct search optimization method that models the objective and constraint functions by linear interpolation. *Pages 51–67 of: Advances in optimization and numerical analysis*. Springer.

SAMUELS, JON D, & SEKKEL, RODRIGO M. 2017. Model confidence sets and forecast combination. *International Journal of Forecasting*, **33**(1), 48–60.

SARTWELL, PHILIP E, *et al.* 1950. The Distribution of Incubation Periods of Infectious Diseases. *American Journal of Hygiene*, **51**(3), 310–18.

THEIL, HENRI. 1971. Principles of Econometrics, New York: JohnWiley. *TheilPrinciples of Econometrics1971*.

TIMMERMANN, ALLAN. 2006. Chapter 4 Forecast Combinations. *Pages 135–196 of: Handbook of Economic Forecasting*. Elsevier.

WEARING, HELEN J, ROHANI, PEJMAN, & KEELING, MATT J. 2005a. Appropriate models for the management of infectious diseases. *PLoS medicine*, **2**(7).

WEARING, HELEN J, ROHANI, PEJMAN, & KEELING, MATT J. 2005b. *Protocol S1. Futher details and Analyis of the Mathematical Models*. 7 KB TEX.

WEIHS, CLAUS, LUEBKE, KARSTEN, & CZOGIEL, IRINA. 2006. *Response surface methodology for optimizing hyper parameters*. Tech. rept. Technical Report.

WEISE, THOMAS. 2009. Global optimization algorithms-theory and application. *Self-Published Thomas Weise*.

WEST, KENNETH D. 1996. Asymptotic inference about predictive ability. *Econometrica: Journal of the Econometric Society*, 1067–1084.