Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

Eivind Baltzersen

Performance prediction with a hierarchical poisson model using Template Model Builder

Master's thesis in Applied Physics and Mathematics
Supervisor: Jarle Tufto
July 2020

NTNU
Norwegian University of Science and Technology

Eivind Baltzersen

Performance prediction with a hierarchical poisson model using Template Model Builder

Master's thesis in Applied Physics and Mathematics
Supervisor: Jarle Tufto
July 2020

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU
Norwegian University of
Science and Technology

This document was written for the web, and has been exported to a tree unfriendly format at a loss of quality. The web version can be found in the attachments; JavaScript may be required if not using Firefox or Safari.

Sammendrag

I dette prosjektet bruker vi en hierariske poisson–log-normal angreps- og forsvars-modell for Eliteserien 2019 til å finne forventet sluttresultat i tabellen. Vi bruker forskjellige autoregressive tidsrekkemodeller til å modellere endring i angreps- og forsvars-parametrene mellom kampene. Modellene viser seg å være dårlige til å predikere invidiuelle kamper, men bedre til å predikere lagenes sluttposisjon i tabellen. Modellene er implementert i TMB, et R/C++-bibliotek, og plottet i Python. Vi ser på noen svakheter ved modellene, og til slutt foreslår vi noen endringer som kan forbedre dem.

Abstract

In this project we use a hierarchical poisson–log-normal attack and defence model for Eliteserien 2019 (The Norwegian primary football competition) in order to predict the expected final results in the standings. We use different autoregressive time series models to model change in attack and defence parameters. We show the models to be unsuitable for predicting individual matches, but better at predicting a teams position in the final standings. The models are implemented using TMB, an R/C++ library, and are plotted in Python. We look at weaknesses of the models, and finally we suggest changes to improve the models.

Contents

List of Figures

List of Tables

Notation and terms

$\overline{e}$	The inverse of e, e^{-1} or $1/e$.
$\cosh(x)$	The hyperbolic cosine function, $\frac{e^x - e^{-x}}{2}$
$[a..b]$	Discrete closed interval, $a, a+1, \ldots, b-1, b$
$\ell(x), L(x)$	(log-)likelihood function
τ	2π
positive/negative numbers	includes zero
strictly positive/negative numbers	excludes zero

1 Introduction

Ranking is a way to compare different objects given their properties. It is ideally transitive, meaning that for n objects, we may label them with $i \in [1..n]$ to order them. An important aspect of competitive sports and games is to rank teams and players in order to sufficiently determine who is to be labeled 1, (also known as the *winner*).

In a sports context, the ranking often depends on a *score*, which is determined based on the performance of the team or player against another team or player. This is achieved through each team or player playing against other teams and players, for instance through a knockout or a knockout tournament. Examples from football[1] in Norway include the knockout cup *NM i fotball*, and the top level round robin tournament is called *Eliteserien*.

To determine the winner of a football match, we simply look at the team with the most goals within two 45 minute halves of a 90 minute match. Some tournaments also have overtime if the teams are tied after 90 minutes, but we will mainly study Eliteserien, which doesn't include overtimes. The winner of a match gets three points, the loser none. A tie awards both teams with one point. The winner of the league is the team with the most points.

We make note of earlier work, such as [3], which attempts an attack-defence model that we will propose here, but with a different hierarchical model. A simple bivariate poisson model (non-hierarchical) has also been attempted in [15]. A model using the skellam distribution has also been attempted in [16]. A comparison of a poisson scoring models with a goal shots model is compared in [24]. There are many other papers for similar models, both in football and other sports and games competitions.

The ideas in this paper are mostly based on earlier works. The novel methods is applying TMB as a tool to implement the models, and comparing different time series models.

As a small disclaimer, I will mention that I have little understanding of football, and have not watched a single football match during the writing of this thesis. So any statements about football *may* be incorrect.

1.1 Tools

The main programming languages have been R and C++, with the TMB library. Results were stored in json format. Plots have been made using Python, with matplotlib, NumPy and SciPy. For more details

2 Theory

We will go through several statistical concepts; it will be assumed that the reader has some knowledge of linear algebra. We will not go through each theorem in depth, nor the derivation of them. For a comprehensive explanation, a textbook should be consulted.

- Distributions
- Models
- Ranking
- Template model builder (TMB)
- Prediction
- Quality measures

[1] Also referred to as association football or soccer.

2.1 Distributions

2.1.1 Normal and log-normal distribution

The normal[2] distribution is a normal distribution, and is described by the density function in equation (1)

$$X \sim f(x;\mu,\sigma^2) = \mathcal{N}(\mu,\sigma^2) = \overline{\sqrt{\sigma^2\tau}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) \mathrm{d}x \tag{1}$$

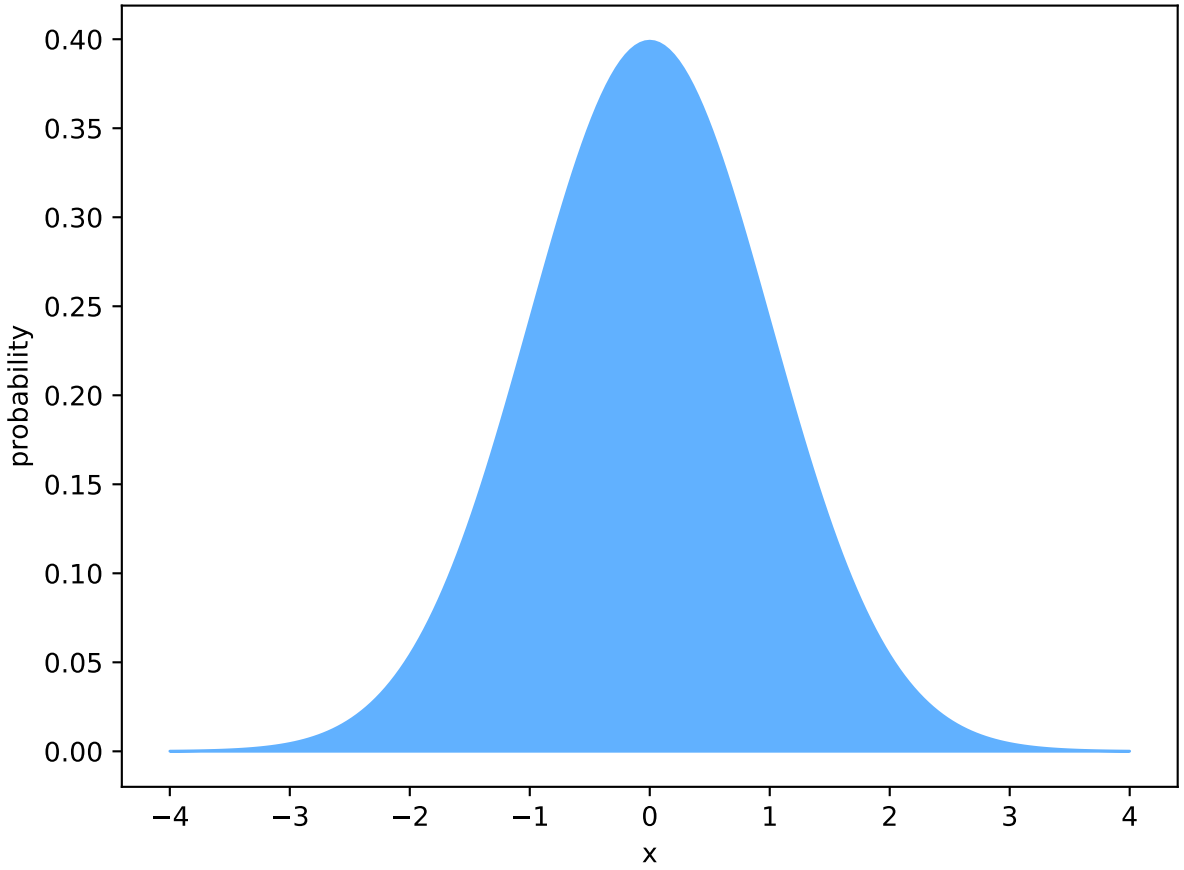

Figure 1: The standard normal distribution $\mathcal{N}(0,1)$ in $[-4,4]$.

The normal distribution is fully parameterised by its mean and variance:

$$E(X) = \mu \tag{2}$$

$$\mathrm{Var}(X) = \sigma^2 \tag{3}$$

With the (unbiased) estimators

$$\hat{\mu} = \sum_{i=1}^{n} \frac{X_i}{n} \tag{4}$$

$$\hat{\sigma}^2 = s^2 = \sum_{i=1}^{n} \frac{X_i - \hat{\mu}}{n-1} \tag{5}$$

$$\hat{\sigma} = c(n)\sqrt{\sum_{i=1}^{n} \frac{X_i - \hat{\mu}}{n-1}} \tag{6}$$

where $c(n)$ is a bias-correction factor, because $E(s) < \sigma$. For large samples this is close to one, so this factor is typically ignored, but exact and approximate values for the normal distribution can be found. [12] [31] [6]

[2] Also known as gaussian distribution, gauss distribution, laplace–gauss distribution, normal distribution, bell curve. It can more accurately be called a quadratic-normal distribution.

The log-normal distribution is similar, with a change of variables $x \to \log(x)$:

$$Y = \exp(X) \sim f(x; \mu, \sigma^2) = \text{Lognormal}(\mu, \sigma^2) = \overline{x\sqrt{\sigma^2\tau}} \exp\left(-\frac{1}{2}\left(\frac{\log(x) - \mu}{\sigma}\right)^2\right) \, \mathrm{d}y \tag{7}$$

We note that $\log(Y)$ is normal distributed, which we will make use of later, as the normal distribution is simpler.

The multivariate form of the normal distribution, often abbreviated to MVN, is

$$\mathbf{X} \sim f(\mathbf{x}; \mu, \Sigma) = \mathcal{N}(\mu, \Sigma) = \overline{\sqrt{|\Sigma|\tau^k}} \exp(-\frac{1}{2}(\mathbf{x} - \mu)^\top \overline{\Sigma}(\mathbf{x} - \mu)) \, \mathrm{d}\mathbf{x} \tag{8}$$

with the parameters

$$E(X) = \mu \tag{9}$$

$$\text{Var}(X) = \Sigma \tag{10}$$

Σ is usually referred to as a covariance matrix.

We will consider a special parametrisation of the covariance matrix. The motivation is to removed restrictions on the entries. First of all, the covariance matrix is symmetric, so almost half of the entries are redundant when specifying the matrix. A more complex restriction, is that it must also be positive definite, i.e. $(x - \mu)^\top \Sigma (x - \mu) \geq 0$ and its diagonal entries are strictly positive.

First we consider the relation between the covariance matrix and the correlation matrix:

$$P = \overline{\text{diag}(\Sigma)}^{1/2} \Sigma \overline{\text{diag}(\Sigma)}^{1/2} \tag{11}$$

where P is the correlation matrix, which have the nice property that its diagonal consists of units. We can find a lower-triangular square root L, such that $P = LL^\top$. However, L does not have a unit diagonal, for this we multiply by the inverse of its diagonal to obtain $\Theta = \overline{\text{diag}(L)} L$, as shown in equation (12)

$$\Theta = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \theta_1 & 1 & 0 & \cdots & 0 \\ \theta_2 & \theta_3 & 1 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 0 \\ \theta_k & \theta_{k+1} & \cdots & \theta_{k+n} & 1 \end{pmatrix} \tag{12}$$

We let θ be the vectorisation of the lower triangular matrix, i.e. $\theta = (\theta_1, \theta_2, \ldots, \theta_{k+n})$.

We may also consider the elementwise root-log transform of the diagonal of Σ:

$$\log(\sigma) = (\log(\sqrt{\sigma_1^2}), \log(\sqrt{\sigma_2^2}), \ldots, \log(\sqrt{\sigma_k^2})) = (\log(\sigma_1), \log(\sigma_2), \ldots, \log(\sigma_k)) \tag{13}$$

Thus, we may parametricise Σ as from the vector $\theta \oplus \log(\sigma) \in \mathbb{R}^{T(k)}$, where $T(k)$ is the k-th triangular number by reversing the above steps. [17]

2.1.2 Binomial distribution and the fundamental theorem of statistics

In determining the outcome of a match, we are either right or wrong. If we have a rule determining the correct outcome of a match with probability p, we have a binary distribution[3].

Repeated determination of n matches, results in a binomial distribution, where we expect the number of the correct guessed outcomes m, such that $m/n \approx p$.

$$f(n, k|p) = (\frac{n}{k}) p^k (1 - p)^{n-k} \tag{14}$$

[3] Also referred to as the bernoulli distribution

The limiting distribution of a binomial variable X_n as $n \to \infty$, can be approximated by a normal distribution. This theorem is sometimes referred to as the de moivre–laplace theorem, which is a special case of the central limit theorem[4].

$$\frac{X_n - np}{\sqrt{np(1-p)}} \overset{\lim}{\sim} N(np, np(1-p)) \tag{15}$$

Typically, this approximation is practical for n greater than 30.

The estimator for p, $\hat{p}$, is simply the number of correct guessed m over the total outcomes. $m/n = \hat{p}$. So the confidence interval, assuming normality, is of the form

$$p \in [\hat{p} - z_\alpha c(n)\sqrt{\hat{p}(1-\hat{p})/n}, \hat{p} + z_\alpha c(n)\sqrt{\hat{p}(1-\hat{p})/n}] \tag{16}$$

We reiterate that this is unreliable for a small sample size, and other confidence intervals also exist. A list of other methods can be found in [36].

2.1.3 Poisson distribution

The poisson distribution is defined as the number of events occuring a fixed interval. It's described by the mass function in equation (17)

$$Y \sim f(y; \lambda) = \text{Pois}(\lambda) = \frac{\lambda^y e^{-\lambda}}{y!} \tag{17}$$

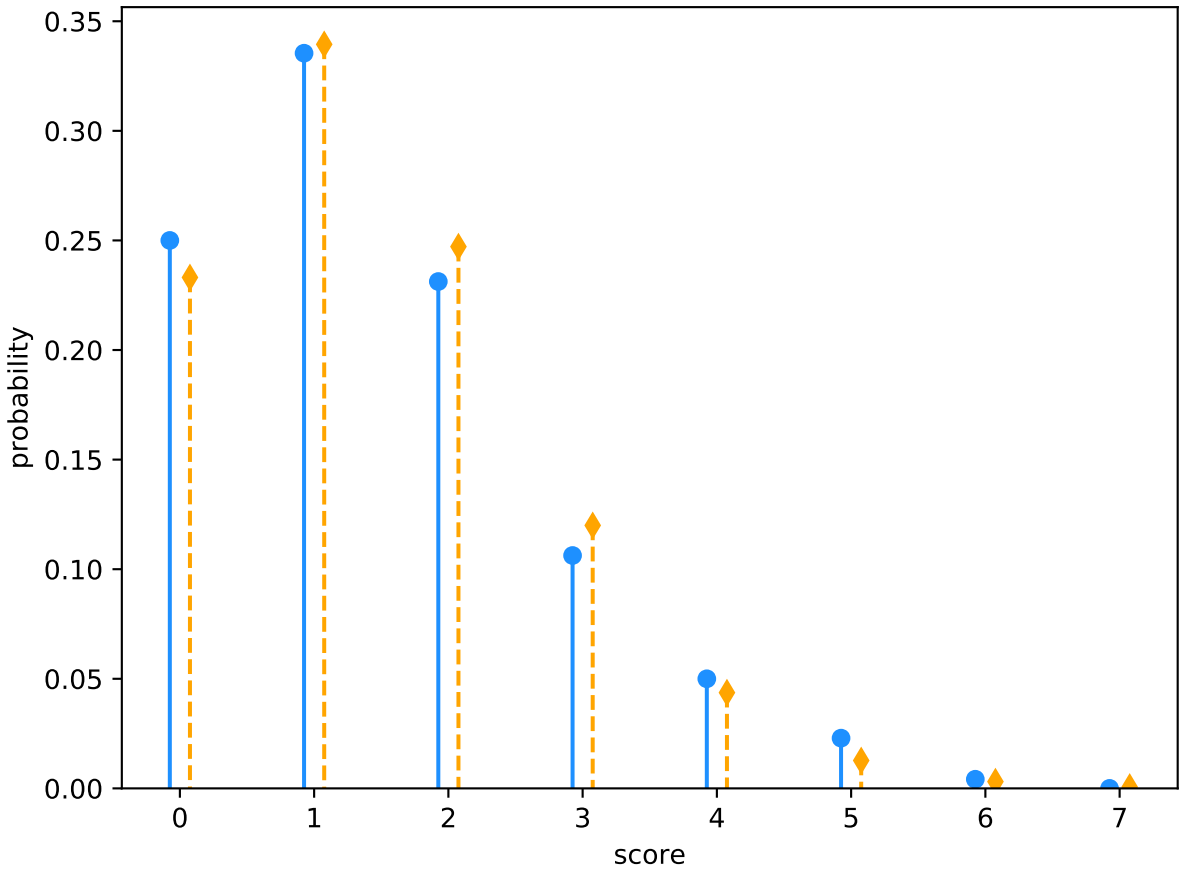

Figure 2: The left solid blue poles with round hats make up the score distribution of Eliteserien 2019. The right orange dashed poles with diamond hats are a fitted poisson distribution with $\lambda \approx 1.46$.

A remarkable property of the poisson distribution, is the simple relation between the mean and variance:

$$E(Y) = \text{Var}(Y) = \lambda \tag{18}$$

[4]The central limit theorem is sometimes referred to as the fundamental theorem of statistics, though the law of large numbers also called by this name.

2.1.4 Skellam distribution

The difference between two independent poisson distributed variables is distributed by the skellam distribution. [32] The skellam distribution is of the form show in equation (19)

$$p(k|\lambda_1, \lambda_2) = e^{-(\lambda_1+\lambda_2)}\left(\frac{\lambda_1}{\lambda_2}\right)^{k/2} I_{|k|}(2\sqrt{\lambda_1\lambda_2}) \tag{19}$$

where I_k is the modified bessel function of the first kind. The formula is complicated, but is shown below [1, page 375]

$$I_\alpha(x) = i^{-\alpha}J_\alpha(ix) = \sum_{m=0}^{\infty} \frac{1}{m!\Gamma(m+\alpha)!}\left(\frac{x}{2}\right)^{2m+\alpha} \tag{20}$$

2.1.5 Discrete VAR(1) model

The VAR(1) series is the multivariate generalization of the one-dimensional AR(1) series, which is a special case of the AR(p) series:

$$x_t = c + \phi_1 x_{t-1} + \cdots + \phi_p x_{t-p} + \epsilon_t \tag{21}$$

where $\epsilon_t \sim \text{WN}(0, \Sigma_\epsilon)$. [5, page 84] In our case, we consider $p = 1$. We will also be assuming that $c = 0$. The distribution is stable as long as $|\phi_1| < 1$.

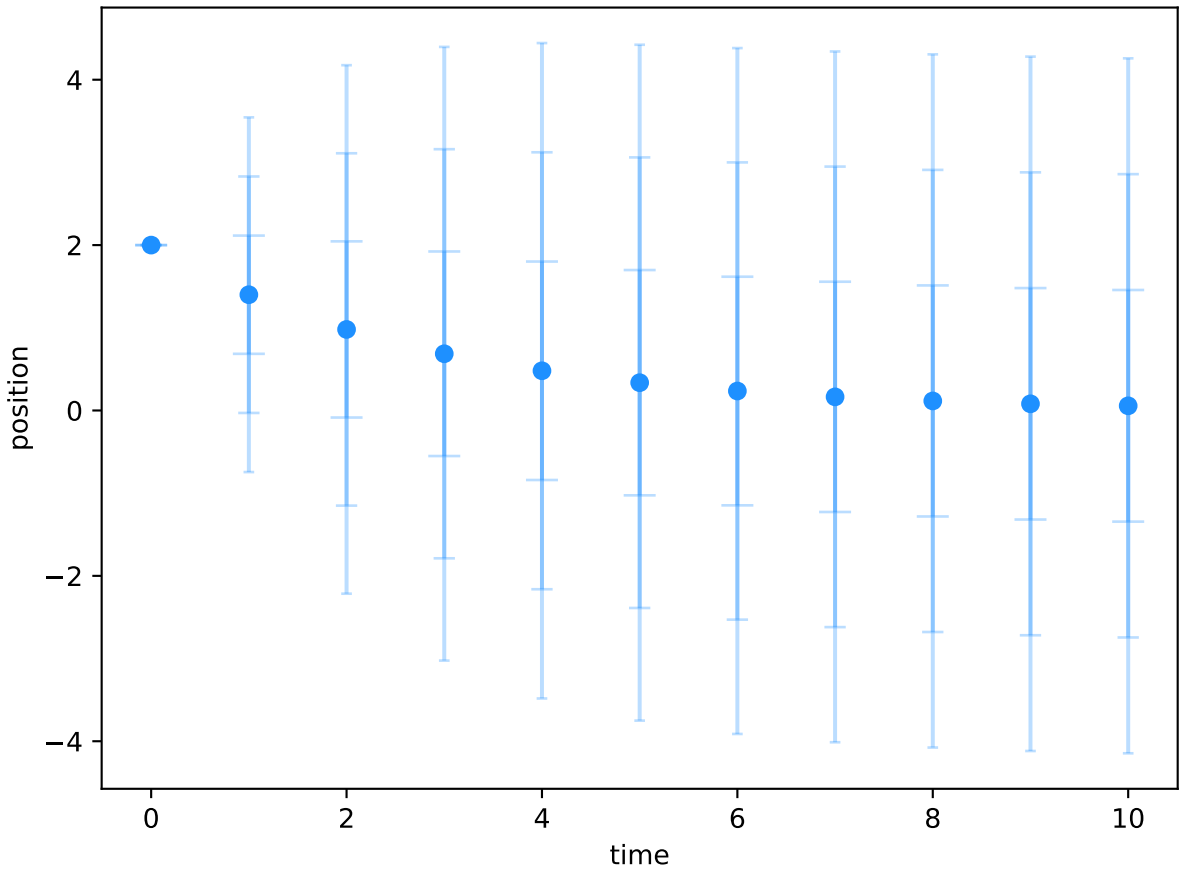

Figure 3: The conditional expectation and confidence interval of a Var(1) process, with $\phi = 0.7$, $\Sigma_\epsilon \approx 0.71$ and $X_0 = 2$. Each interval at each point represents one, two and three standard deviations from the expected mean.

The unconditional expectation and covariance are given by [10]

$$E(x_t) = \frac{c}{1-\phi} = \mu \tag{22}$$

$$\text{Var}(x_t) = \frac{\Sigma_\epsilon}{1-\phi^2} \tag{23}$$

and the unconditional distribution of x_t is

$$x_t \sim \mathcal{N}(\mu, \frac{\Sigma_\epsilon}{1-\phi^2}) \tag{24}$$

The conditional expectation and covariance are given by

$$E(x_t|x_{t-1}) = c + \phi x_{t-1} \tag{25}$$

$$\mathrm{Var}(x_t|x_{t-1}) = \Sigma_w \tag{26}$$

and the conditional distribution of x_t on x_{t-1} is given by

$$x_t|x_{t-1} \sim \mathcal{N}(c + \phi x_{t-1}, \Sigma_w) \tag{27}$$

The multivariate version of the unconditional and conditional distribution is: [14]

$$\mathbf{x}_t \sim \mathcal{N}(\mu, \overline{\mathrm{vec}}(\overline{I - \phi_1 \otimes \phi_1}\mathrm{vec}(\Sigma_w))) \tag{28}$$

$$\mathbf{x}_t|\mathbf{x}_{t-1} \sim \mathcal{N}(c + \phi \mathbf{x}_{t-1}, \Sigma_w) \tag{29}$$

We will later be using equation (28) and equation (29), with zero-shifted mean.

2.1.6 Continuous VAR model

The continuous version[5] is often attributed to Ornstein, Uhlenbeck and Vašíček. [35] [28] The differential form and the formal solutions are shown in equation (30).

$$\mathrm{d}x_t = \theta(\mu - x_t)\mathrm{d}t + \sigma_w \mathrm{d}W \tag{30}$$

$$x_t = x_0 e^{-\theta t} + \mu(1 - e^{-\theta t}) + \sigma_w \int_{s=0}^{t} e^{-\theta(t-s)} \mathrm{d}W_s \tag{31}$$

where the absolute value of the eigenvalues of θ should be strictly positive in the real part to be stable. [4, page 11] The relation between these two form can be found in the appendix, in equation (120)

The unconditional expectation and covariance are given by [10]

$$E(x_t) = \mu \tag{32}$$

$$\mathrm{Var}(x_t) = \frac{\sigma_w^2}{2\theta} \tag{33}$$

We will be assuming $\mu = 0$, but we state the general forms for completeness.

and the unconditional distribution of x_t is [9]

$$x_t \sim \mathcal{N}(\mu, \frac{\sigma_w^2}{2\theta}) \tag{34}$$

The conditional expectation and covariance are given by [10]

$$E(x_t|x_{t-1}) = e^{-\theta\Delta t} x_0 + \mu(1 - e^{-\theta\Delta t}) \tag{35}$$

$$\mathrm{Var}(x_t|x_{t-1}) = \frac{\sigma_w^2}{2\theta}(1 - e^{-2\theta\Delta t}) \tag{36}$$

and the conditional distribution of x_t on x_{t-1} is given by [33, page 11]

$$x_t|x_{t-1} \sim \mathcal{N}(e^{-\theta\Delta t} x_{t-1} + \mu(1 - e^{-\theta\Delta t}), \frac{\sigma_w^2}{2\theta}(1 - e^{-2\theta\Delta t})) \tag{37}$$

[5] Often referred to as the Ornstein–Uhlenbeck process in literature.

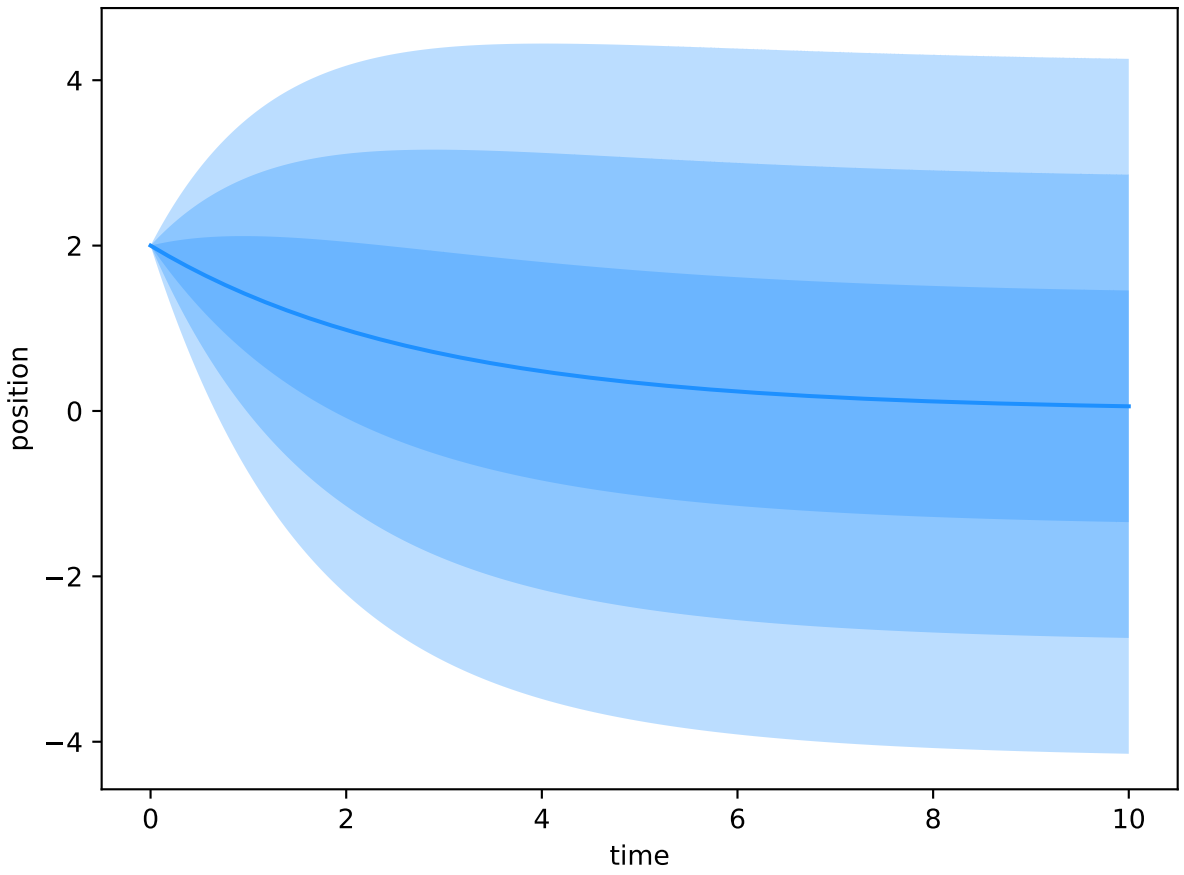

Figure 4: The conditional expectation and confidence interval of a Var(1) process, with $\theta = -\log(0.7) \approx 0.35$, $\Sigma_\epsilon = 1$ and $x_0 = 2$. Each band represents one, two and three standard deviations from the expected mean. This is the continuous extension of figure 3.

The multivariate version of the unconditional and conditional distribution is: [37]

$$\mathbf{x}_t \sim \mathcal{N}(\mu, \overline{\text{vec}}(\overline{\theta \oplus \theta}\text{vec}(\sigma_w \sigma_w^\top))) \tag{38}$$

$$\mathbf{x}_t | \mathbf{x}_{t-1} \sim \mathcal{N}(e^{-\theta \Delta t}\mathbf{x}_{t-1} + \mu(I - e^{-\theta \Delta t}), \overline{\text{vec}}(\overline{\theta \oplus \theta}(I - e^{-\theta \oplus \theta \Delta t})\text{vec}(\sigma_w \sigma_w^\top))) \tag{39}$$

As with the VAR(1) process, we will later be using equation (38) and equation (39), with zero-shifted mean.

The relation to the VAR(1) model is not immediately obvious, but equality can be shown with the following substitutions: [25][11, page 8]

$$X_0 = x_0 \tag{40}$$

$$\varphi = e^{-\theta \Delta t} \tag{41}$$

$$\epsilon_t \sim \mathcal{N}(0, \frac{1}{2\theta}\sigma_w^2(1 - e^{-2\theta \Delta t})) \tag{42}$$

so the following two equations are equivalent

$$x_t = e^{-\theta \Delta t}x_{t-1} + \sigma_w \int_{s=0}^{t} e^{-\theta(t-s)} \mathrm{d}W_s \tag{43}$$

$$X_t = \varphi X_{t-1} + \epsilon_t \tag{44}$$

2.2 Change of variables in a density function

In some cases it may be useful to transform the variables, such as $Y = g(X)$, because we may have more tools available for the transformed density. Such as log-transforming a log-normal variable, to obtain a normal

variable.

$$f_Y(y)\,\mathrm{d}y = f_{\overline{g}(Y)}(\overline{g}(y))\mathrm{d}(\overline{g}(y)) = f_X(x)\,\mathrm{d}x \tag{45}$$

To be precise, this is only valid as long as g is a strictly increasing function. For the decreasing case we add a sign to either side; the two cases can be unified by applying the absolute value to both sides. [26]

and the relation between f_Y and f_X is given by

$$f_Y(y) = f_X(\overline{g}(y))\frac{\mathrm{d}}{\mathrm{d}y}\overline{g}(y) \tag{46}$$

If we let $Y = \exp(X) \sim \text{Lognormal}(\mu, \sigma^2)$, then

$$f_Y(y) = \frac{\mathrm{d}}{\mathrm{d}y}\log(y)\overline{\sigma\sqrt{\tau}}\exp\left(-\frac{1}{2}\left(\frac{\log(y)-\mu}{\sigma}\right)^2\right)\,\mathrm{d}y \tag{47}$$

$$= \frac{1}{y}\overline{\sigma\sqrt{\tau}}\exp\left(-\frac{1}{2}\left(\frac{\log(y)-\mu}{\sigma}\right)^2\right)\,\mathrm{d}y \tag{48}$$

And for the opposite case, $X = \log(Y) \sim \mathcal{N}(\mu, \sigma^2)$, then

$$f_X(x) = \frac{\mathrm{d}}{\mathrm{d}x}\exp(x)\frac{1}{\exp}(x)\overline{\sigma\sqrt{\tau}}\exp\left(-\frac{1}{2}\left(\frac{\log(\exp(x))-\mu}{\sigma}\right)^2\right)\,\mathrm{d}x \tag{49}$$

$$= \frac{\exp(x)}{\exp(x)}\overline{\sigma\sqrt{\tau}}\exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)\,\mathrm{d}x \tag{50}$$

$$= \overline{\sigma\sqrt{\tau}}\exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)\,\mathrm{d}x \tag{51}$$

2.3 Likelihood function

The most general form of the likelihood function is defined as a mass/density function of a parameter θ given an outcome x

$$L(\theta|x) = p_\theta(x) \tag{52}$$

In many cases, x is a tuple of i.i.d. variables, and $p_\theta(x)$ is a joint probability distribution of independent variables, and can be written as a product. So if x is a vector of size n, we have:

$$L(\theta|x) = \prod_{i=1}^{n} p_\theta(x_i) \tag{53}$$

2.3.1 Maximization and log-likelihood

The likelihood function represents the probability of obtaining x for a given θ. A reasonable assumption is then that x is realized from a distribution where it has a high likelihood to be observed. So the θ that yields the highest likelihood is a natural candidate for determining the distribution of x. We seek to determine

$$\hat{\theta} = \underset{\theta\in\Theta}{\operatorname{argmax}} L(\theta|x) = \underset{\theta\in\Theta}{\operatorname{argmax}} \prod_{i=1}^{n} p_\theta(x_i) \tag{54}$$

Product are often cumbersome, and to simplify the above computation, we can apply the logarithm to L to obtain a sum. Because the logarithm is a strictly increasing function, the maximum of L is also the maximum of $\log(L)$. We denote this logarithm by ℓ:

$$\hat{\theta} = \underset{\theta\in\Theta}{\operatorname{argmax}} \ell(\theta|x) = \underset{\theta\in\Theta}{\operatorname{argmax}} \sum_{i=1}^{n} p_\theta(x_i) \tag{55}$$

From which we conclude that x most likely derived from the distribution $p(x|\hat{\theta})$.

2.4 Hierarchical modelling and empirical bayes method

Bayesian[6] hierarchical modelling is a way to describe a hierarchy of distributions, where the parameters of the upper layers are dependent on the distributions of lower layers.

The simplest such model is the two-stage hierarchical model, as shown in equation (56)

$$Y|\theta,\phi \sim p(Y|\theta,\phi) \tag{56}$$

$$\theta|\phi \sim p(\theta|\phi) \tag{57}$$

$$\phi \sim p(\phi) \tag{58}$$

where Y is the observed data, θ is a parameter, and ϕ is a hyperparameter. $p(\theta,\phi)$ is the prior distribution. In bayesian hierarchical modelling the hyperparameter is given a *hyperprior*, a distribution on the parameter, or $p(\phi)$.

In Empirical Bayes the hyperparameter is a fixed value. The estimation of this hyperparameter will be found using MLE.

The **posterior theorem** or **bayes theorem** is shown below in equation (59)

$$f(\theta|x) = \frac{f(x|\theta)f(\theta)}{f(x)} = \frac{f(x|\theta)f(\theta)}{\sum_i f(x|\theta_i)f(\theta_i)} \tag{59}$$

The distribution for Y can then be found by marginalization[7] over the parameters, or random effects.

$$p(Y|\phi) = \int_{\mathbb{R}} p(Y|\theta,\phi)p(\theta|\phi)\mathrm{d}\theta = \int_{\mathbb{R}} \frac{p(\theta|Y,\phi)p(Y|\phi)}{\sum_i p(\theta|Y_i,\phi)p(Y_i)} p(\theta|\phi)\mathrm{d}\theta \tag{60}$$

2.5 Bradley–terry model

The bradley–terry model [29] is used to make paired comparisons of individuals in a transitive way, using a single parameter.

$$P(i > j) = \frac{p_i}{p_i + p_j} = \sigma(\beta_i - \beta_j) \tag{61}$$

Where $p_i = e^{\beta_i}$ and $\sigma(x) = \overline{1 - \exp(x)}$ is the logistic function (a sigmoid function). The relation between the $\mathrm{logit}(x) = \log(\frac{x}{1-x})$ logit function with the logistic function is that they are inverses, i.e. $\mathrm{logit}(x) = \overline{\sigma}(x)$, so we also have the identity:

$$\mathrm{logit}P(i < j) = \mathrm{logit}(\sigma(\beta_i - \beta_j)) = \beta_i - \beta_j \tag{62}$$

An example where this model is used, is in elo ranking, most commonly known from chess. A player's rating is given by a number r_i, which is related to β_j by $\beta_i = \log(10)/400 \cdot r_i$. So a player with $r_1 = 1700$ playing against a player with $r_2 = 1900$ will yield the following probabilities:

$$P(1 > 2) = \overline{1 - 10^{(1900-1700)/400}} \approx 25\% \tag{63}$$

$$P(1 < 2) = \overline{1 - 10^{(1700-1900)/400}} \approx 75\% \tag{64}$$

A win or a loss updates the elo rating of each player, but the bradley–terry model has no mechanism for updating. Nor does it tell us how to initially calculate ratings, which must be found with inference.

We also note that the bradley–terry model does not handle ties, only binary win/loss outcomes.

The winner is the team with the highest score in a match. If the score is poisson distribution, we can determine the result by checking the value of $y_{ij}^{\mathrm{Home}} - y_{ij}^{\mathrm{Away}}$, and checking if it is strictly positive, zero or strictly negative.

2.5.1 Skellam distribution and the bradley–terry model

By defining the best team as having the score $p_M = 1$, we could use this to solve

$$P(i < j) = \frac{p_i}{p_i + p_M} \tag{65}$$

for any p_i to obtain a ranking score for all teams. Instead we will use the point system.

[6] The naming is unfortunate, but refers to Thomas Bayes. It may be more intuitive to think of it as evidence-based.

[7] Integrating out the distributions of variables.

2.6 Template model builder (TMB, R library)

Template model builder is an R/Cpp-library which greatly simplifies the means of estimating hierarchical models. [23][22][18]. It makes use of three key concepts: *automatic differentiation*, *laplace method* and the (generalized) *delta method*.

Understanding these concepts are not necessary to make use of the program, but they can be helpful.

2.6.1 Automatic differentiation and dual numbers

Dual numbers are 2-dimensional vectors with the unit vectors 1 and ϵ denoted by (a, b) or $a+b\epsilon$, with the added property that the square of the second component is identified with 0. [38, page 41-43]

$$\epsilon^2 = 0 \tag{66}$$

This has applications for differentiation, where it can be used to find derivatives without differentiating. We define two dual vectors[8] $\mathbf{u} = (u, u')$ and $\mathbf{v} = (v, v')$.

$$\mathbf{u} + \mathbf{v} = (u, u') + (v, v') = (u + v, u' + v') \tag{67}$$

$$\mathbf{u} - \mathbf{v} = (u, u') - (v, v') = (u - v, u' - v') \tag{68}$$

$$\mathbf{uv} = (u, u')(v, v') = (uv, u'v + uv') \tag{69}$$

$$\frac{\mathbf{u}}{\mathbf{v}} = \frac{(u, u')}{(v, v')} = \left(\frac{u}{v}, \frac{u'v - uv'}{v^2}\right) \tag{70}$$

$$f(\mathbf{u}) = f(u, u') = (f(u), f'(u)u') \tag{71}$$

We note the similarity between the second component and the rules from differentiation. While the first four rules are trivial, the last requires an explanation. This is found by the tangent expansion[9] of the function:

$$f(\mathbf{u}) = f(u + u'\epsilon) = \sum_{i=0}^{\infty} \frac{f^{(n)}(u)(u'\epsilon)^n}{n!} = f(u) + f'(u)u'\epsilon \tag{72}$$

The chain rule is also applicable:

$$f(g(\mathbf{u})) = f(g((u, u'))) = f((g(u), g'(u)u')) = (f(g(u)), f'(g(u))g'(u)u') \tag{73}$$

By mapping a variable to $x \to (x_0, 1)$ and a constant to $c \to (c, 0)$, any[10] function can be differentiated by applying the chain rule until sufficiently elementary functions can be computed in order.

For a thorough introduction to the automatic differentiation, we refer to the paper of the stan math library. [7] However, this is only useful for understanding the underlying math of TMB, not for using TMB, so it can safely be ignored.

2.6.2 Laplace method

The second tool is the laplace method, which helps us approximate the marginal distribution and estimate the mean of the fixed parameters and random effects.

The laplace method is based off the tangent series and a quadratic-exponential integral identity[11].

$$\int_{-\infty}^{\infty} e^{-a(x+b)^2} \, \mathrm{d}\theta = \sqrt{\frac{\pi}{a}} \tag{74}$$

We also assume that $f(\theta, u)$ achieves its maximum at $\hat{u}$, i.e. $\partial_u f(\theta, \hat{u}) = 0$, and that $a = b = \infty$ (or that the function decays sufficiently fast from $\hat{\theta}$). And last, we assume that it achieves it's peak at $\hat{u}$, i.e. $\partial^2_{uu} f(\theta, \hat{u}) < 0$.

[8] Note that this is unrelated to the concept of dual spaces.
[9] More commonly known as a taylor series.
[10] Any, meaning suitably nice.
[11] More commonly known as the gaussian integral, or the euler–poisson integral

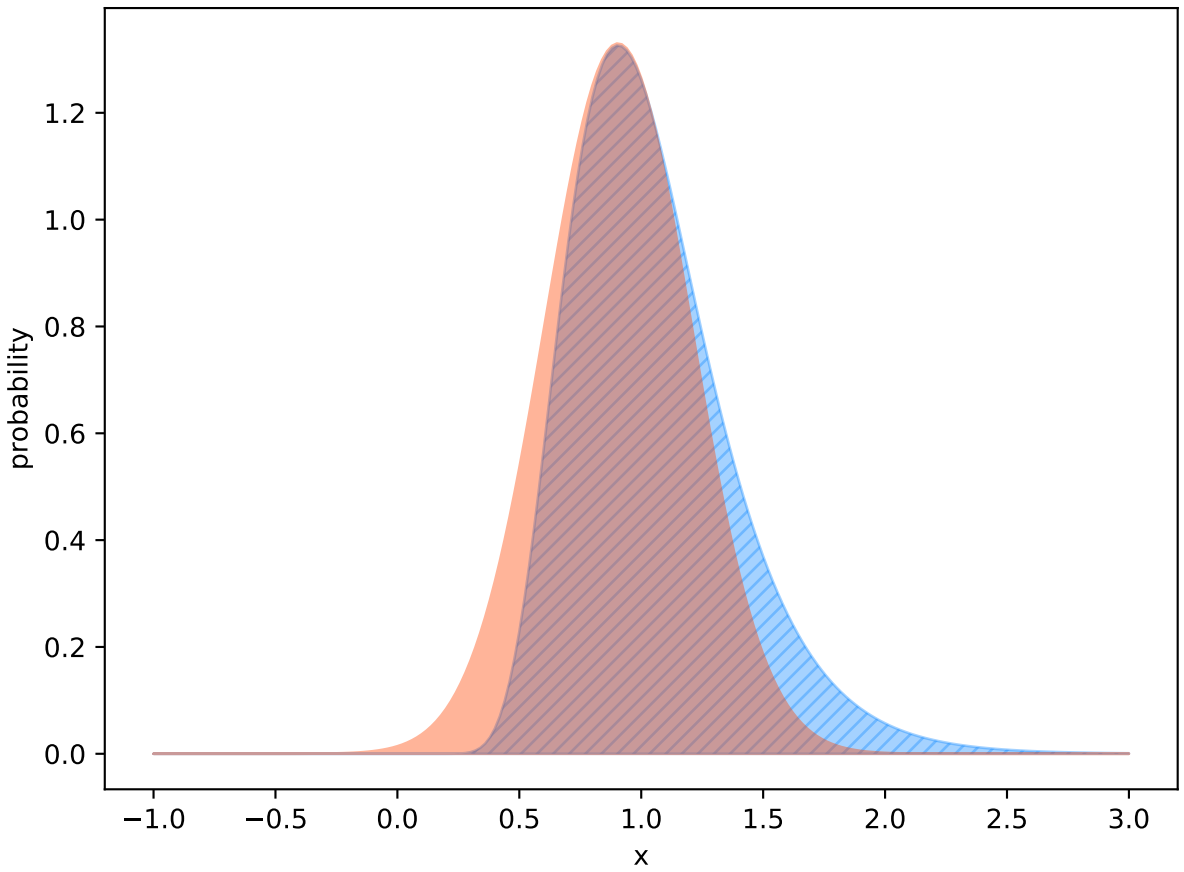

Figure 5: A normal approximation of a Lognormal$(0, 0.1^2)$ distribution. The blue dashed area is the log-normal distribution, and the red plain area is the normal approximation. This was not computed using the method in this section, but the idea is similar.

The proof then follows.

$$L(\theta) = \int_{x=a}^{b} L(\theta|u)\,\mathrm{d}u \tag{75}$$

$$= \int_{x=a}^{b} \exp(M(\overline{M}\overline{\exp}(L(\theta|u))))\,\mathrm{d}u \tag{76}$$

$$= \int_{x=a}^{b} \exp(Mf(\theta,u))\,\mathrm{d}u \tag{77}$$

$$\approx \int_{x=a}^{b} \exp(M(f(\theta,\hat{u}) + \cancel{\partial_u f(\theta,\hat{u})(u-\hat{u})} + \frac{1}{2}\partial^2_{uu} f(\theta,\hat{u})(u-\hat{u})^2))\,\mathrm{d}u \tag{78}$$

$$= \int_{x=a}^{b} \exp(M(f(\theta,\hat{u}) + \frac{1}{2}\partial^2_{uu} f(\theta,\hat{u})(u-\hat{u})^2))\,\mathrm{d}u \tag{79}$$

$$= \exp(Mf(\theta,\hat{u}))\int_{x=a}^{b} \exp(\frac{1}{2}M\partial^2_{uu} f(\theta,\hat{u})(u-\hat{u})^2)\,\mathrm{d}u \tag{80}$$

$$= \exp(Mf(\theta,\hat{u}))\int_{x=a}^{b} \exp(-\frac{1}{2}M|\partial^2_{uu} f(\theta,\hat{u})|(u-\hat{u})^2)\,\mathrm{d}u \tag{81}$$

$$\overset{\lim}{=} \exp(Mf(\theta,\hat{u}))\int_{-\infty}^{\infty} \exp(-\frac{1}{2}M|\partial^2_{uu} f(\theta,\hat{u})|(u-\hat{u})^2)\,\mathrm{d}u \tag{82}$$

$$= \exp(Mf(\theta,\hat{u}))\sqrt{\frac{\pi}{\frac{1}{2}M|\partial^2_{uu} f(\theta,\hat{u})|}} \tag{83}$$

$$= \exp(Mf(\theta,\hat{u}))\sqrt{\frac{2\pi}{M|\partial^2_{uu} f(\theta,\hat{u})|}} \tag{84}$$

$$= \exp(Mf(\theta,\hat{u}))\sqrt{\frac{\tau}{-M\partial^2_{uu} f(\theta,\hat{u})}} \tag{85}$$

We will use the special case of $M = -1$, thus $f(\theta, u) = -\log(L(\theta, u))$ and

$$-\log(L(\theta)) \approx -\log(L^*(\theta)) = -\log(\sqrt{\tau}) + \frac{1}{2}\log(\partial^2_{uu} f(\theta, \hat{u})) + f(\theta, \hat{u}) \tag{86}$$

The multivariate form is slightly different. [23, equation 4] TMB uses this to approximate the posterior distribution.

A related result is the posterior central limit theorem[12]. We leave out the details and the assumptions, but state the result, often attributed to Bernstein and Von Mises: [34] [20]

$$p(u|\theta) \approx \mathcal{N}(\hat{u}, -\overline{n\mathscr{I}(\hat{u})}) \tag{87}$$

where $\mathscr{I}(\hat{u}) = \overline{n}\partial^2_{uu}\log(p(u|\theta))$ is the observed information[13] (i.e. $E(\mathscr{I}) = \mathcal{I}$, where $\mathcal{I}$ is called the information) and n is the number of observations.

2.6.3 Delta method

The third tool is the (generalized) delta method,[8, page 240-243] which helps us estimate the standard deviation of the fixed parameters and random effects.

The regular delta method states that if there exists a sequence of random variables X_n such that

$$\sqrt{n}(X_n - \theta) \xrightarrow{D} \mathcal{N}(0, \sigma^2) \tag{88}$$

where $\xrightarrow{D}$ denotes convergence in distribution, then for a differentiable function g, then

$$\sqrt{n}(g(X_n) - g(\theta)) \xrightarrow{D} \mathcal{N}(0, g'(\theta)^2\sigma^2) \tag{89}$$

The method used in TMB is a variant that approximates the distribution using the laplace method.

If the posterior distribution of $\lambda|y$ is asymptotically normal with mode $\tilde{\lambda}$, then

$$E(g(\lambda)) = g(\tilde{\lambda}) + \mathcal{O}(n^{-1}) \tag{90}$$

$$\mathrm{Var}(g(\lambda)) = g'(\tilde{\lambda})\Sigma g'(\tilde{\lambda})^\top + \mathcal{O}(n^{-2}) \tag{91}$$

where n is the number of observations and $\mathrm{Var}(\lambda) \approx \Sigma = \overline{-\partial^2_\lambda \log(p(\lambda = \tilde{\lambda}|y))}$ is the "inverse of the negative hessian of the log posterior". Instead of the posterior, one may substitute this for $L(\lambda|y) \times p(\lambda)$, as the factor $p(y)$ is constant (and thus its derivative zero). [19]

The more general form where $g(\theta, u)$ also depends on the random effects u, TMB uses a more general estimate:

$$\mathrm{Var}(g(\hat{\theta}, \hat{u})) \approx g'(\hat{\theta}, \hat{u}) \left(\begin{pmatrix} \overline{H} & 0 \\ 0 & 0 \end{pmatrix} + J\mathrm{Var}(\hat{\theta})J^\top \right) g'(\hat{\theta}, \hat{u})^\top \tag{92}$$

where $H = \partial^2_{uu} f(\hat{u}(\theta), \theta)$, the random effects part of the hessian of the objective function, and $J = D_\theta(\hat{u}(\theta), \theta)$, the jacobian of $(u(\theta), \theta)$ wrt. θ. [21]

2.6.4 Bugs

TMB does have some quirks and unexpected behaviour. We list some of those here:

- Bad naming: The negative log-distributions are just called distributions. So evaluating MVNORM yields the negative logarithm of a normal distribution, not the normal distribution.
- Mismatched interfaces: The vectors and matrixes are based off the Eigen library [13], but the arrays are custom made for TMB, and have an incomplete interface compared to vectors and matrixes.
- Unused data and parameter values are silently ignored.

These are practically non-issues, which have simple workarounds, but which one should be aware of. TMB is mature enough to have practical applications, as we demonstrate.

[12] Referred to as the bayesian central limit theorem or the bernstein–von mises theorem in literature.

[13] The observed (fisher) information is typically denoted with a chancery/ca(l)ligraphic J, the use of a spencerian/script I here is just to inconvenience Unicode, which (still) conflates the two fonts.

2.7 Attack and defence model

The model we will be using will be a two-stage empirical poisson-log-prior hierarchical distribution. Read that twice. We will be using different distributions on the priors, but they will look similar.

The hierarchical model is shown below

$$y_{ijt} \sim \text{Poisson}(\lambda_{ijt}) \tag{93}$$

$$\log(\lambda_{ijt}) = \alpha_{it} - \beta_{it} + \gamma x_{ij} + \mu \tag{94}$$

- i and j are team indexes.
- t is the round index.
- y_{ijt} is the number of goals team i scores against j in round j.
- λ_{ijt} is the score parameter for the score.
- μ is the log-average number of goals overall.
- γ is the home advantage. μ and γ are also called *fixed effects*.[14]
- α and β are the attack and defence parameters, respectively. We also refer to these as *random effects* or latent random variables.[15] They attempt to model how each team perform against each other.

Before we describe the priors for the different models, it's important to note the strengths and weaknesses of this model, so we can set realistic expectations of the model performance.

- The model is simple and intuitive to understand.
- It captures the performance of each team individually for every match.
- There isn't a trend parameter; one might expect a team to improve over a season.
- The average and advantage parameters are shared between all teams over all seasons. One would prerably wnat to model each team/season with their separate parameters.

Some of these issues may be resolved by modifying the model, but with a limited data set, and no way to produce more, it's important to keep the model simple to prevent too much overfitting.

Now, for the priors, we make the assumption that these are discretely Var(1) distributed, or continuously VAR distributed.

2.7.1 Time independent model

The simplest model is letting the team parameters be constant throughout the season. We will not study this model in detail, but we mention it.

$$\begin{pmatrix} a_i \\ b_i \end{pmatrix} = w_i \tag{95}$$

where $w_i \sim N(0, \Sigma)$.

2.7.2 Discrete models

In the discrete case we can write the general function

$$\begin{pmatrix} a_i \\ b_i \end{pmatrix}_t = \Phi \begin{pmatrix} a_i \\ b_i \end{pmatrix}_{t-1} + w_t \tag{96}$$

where the absolute value of the eigenvalues of Φ are smaller or equal than 1, and $w_t \sim \text{WN}(0, \Sigma)$. In the initial case (unconditional case), we have

$$\begin{pmatrix} a_i \\ b_i \end{pmatrix}_0 = w_0^* \tag{97}$$

where $w_0^* = \mathcal{N}(0, \Sigma^*)$, with $\Sigma^* = \overline{\text{vec}}(\overline{I - \phi_1 \otimes \phi_1}\text{vec}(\Sigma_w))$ in equation (28).

We consider three cases for conditions of ϕ:

[14] The term vary between bayesian/evidential and frequenist statistics.
[15] See the previous footnote.

- $\Phi = 0$: The terms in the sequence are independent. This is known as a *white noise* process, or more specifically a gaussian white noise, as ϵ are normal distributed. SO this case degenerates to a multivariate normal distribution.
- $\Phi = 1$: The next term is the previous term plus a random step. This is known as a *random walk*.
- Unrestricted. This is a general VAR(1) model. We consider it to be stable as long as the eigenvalues of Φ are strictly smaller than 1.

2.7.3 Continuous models

In the continuous case we can write the general function

$$\begin{pmatrix} a_i \\ b_i \end{pmatrix}_t = e^{-\theta \Delta t} \Phi \begin{pmatrix} a_i \\ b_i \end{pmatrix}_{t-1} + \sigma_w \int_{s=0}^{t} e^{-\theta(t-s)} \mathrm{d}W_s \tag{98}$$

where the absolute value of the eigenvalues of θ should be strictly positive in the real part to be stable. In the initial case (unconditional case), we have

$$\begin{pmatrix} a_i \\ b_i \end{pmatrix}_0 = \sigma_w \int_{s=-\infty}^{0} e^{-\theta(-s)} \mathrm{d}W_s \tag{99}$$

where the random integral has the distribution of $\mathcal{N}(0, \Sigma^*)$, with $\Sigma^* = \overline{\mathrm{vec}}(\overline{\theta \oplus \theta} \mathrm{vec}(\sigma_w \sigma_w^\top))$ as in equation (38).

We consider two cases for conditions of θ:

- $\theta = 0$: The next term is the previous term plus a random step of a given length. This is known as a *wiener process*, which is a continuous extension of a random walk, so we refer to this as a continuous random walk.
- Unrestricted. This is a general VAR(1) model. We consider it to be stable as long as the eigenvalues of θ should be strictly positive in the real part to be stable.
- $\theta \to \infty$ degenerates to a white noise process as in the discrete case, so we ignore this one.

2.8 Result score

2.8.1 Match

While the model itself describse the distribution of the score of a single team, that alone won't help us decide the match winner. For each match we sample from two poisson distributions, or from one skellam distribution.

As usual, the winner is the team with the highest score. If the score of each team is drawn from two poisson distributions, we get y_{ti} and y_{tj}. It is then a simple matter of comparing the two, i.e. check which condition holds in $y_{ti} \gtreqless y_{tj}$.

The equivalent condition in terms of the skellam distribution, is to define $k_{tij} = y_{ti} - y_{tj}$ and check $k_{tij} \gtreqless 0$. So either of these may be used to determine the winner.

Winning a match gives the winning team three points, and the loser none. A tie gives each team one point. This system is known as three points for a win, and is common in football.

2.8.2 Season

Each team plays against every other team twice, in a double round robin system. If there are n teams, then each team plays $2(n-1)$ matches. The score for each match is added up to a final score, from which the overall seasonal winner is determined.

2.9 Likelihood function

After setting up the model, we want to find the optimal parameters

$$L(\gamma, \mu, \Sigma | Y, \lambda, \alpha, \beta) = P(Y, \lambda, \alpha, \beta | \gamma, \mu, \Sigma) \tag{100}$$

$$= P(Y | \lambda, \alpha, \beta, \gamma, \mu, \Sigma) P(\lambda, \alpha, \beta | \Sigma) \tag{101}$$

$$= P(Y | \lambda, \alpha, \beta, \gamma, \mu, \Sigma) P(\alpha, \beta | \Sigma) \tag{102}$$

$$= \prod_{i=0}^{n} P(y_i | \lambda, \alpha, \beta, \gamma, \mu, \Sigma) P(\alpha_0, \beta_0, \Sigma^*) \prod_{j=1}^{m} P(\alpha_j, \beta_j | \alpha_{j-1}, \beta_{j-1}, \Sigma) \tag{103}$$

$$= \prod_{i=0}^{n} P(y_i | \lambda_i, \alpha_i, \beta_i, \gamma, \mu, \Sigma) \phi_{\Sigma^*}(\alpha_0, \beta_0) \prod_{j=1}^{m} \phi_{\Sigma}(\alpha_j, \beta_j | \alpha_{j-1}, \beta_{j-1}) \tag{104}$$

In the independent case the product of the priors reduce to $\prod_{j=0}^{m} P(\alpha_j, \beta_j | \Sigma)$.

With the accompanying log-likelihood:

$$\ell_{\text{Poisson}}(\lambda, \gamma, \mu, \Sigma | Y, \alpha, \beta) = \log(P(Y, \alpha, \beta | \lambda, \gamma, \mu, \Sigma)) \tag{105}$$

$$= \sum_{i=1}^{n} \log(P(y_i | \lambda_i, \alpha_i, \beta_i, \gamma, \mu, \Sigma)) + \sum_{j=1}^{m} \log(\phi_{\Sigma}(\alpha_j, \beta_j)) \tag{106}$$

By using TMB to apply the laplace approximation to the log-likelihood of the model, we obtain a function of $(\mu, \sigma, \phi, \Sigma)$ (or θ in the continuous case), which we can maximize[16]. By maximization we obtain $(\hat{\mu}, \hat{\sigma}, \hat{\phi}, \hat{\Sigma})$, the arguments maximizing the likelihood, and $(\hat{\alpha}, \hat{\beta})$, the mode of the posterior, which we use to find the expected value for λ_i for team i.

2.10 Measuring model quality

2.10.1 Average score parameter

The average of the score parameters for the matches given time-independent parameters is given by

$$\text{avg}(\lambda_j) = \frac{1}{n-1} \sum_{i=1, i \neq j}^{n} \lambda_i \tag{107}$$

$$= \frac{1}{n-1} \sum_{i=1, i \neq j}^{n} \exp(\log(\lambda_i)) \tag{108}$$

$$= \frac{1}{n-1} \sum_{i=1, i \neq j}^{n} \exp(\mu + x_i \gamma + \alpha_i + \beta_i) \tag{109}$$

$$= \frac{1}{n-1} \exp(\mu + \alpha_j) \cosh(\gamma) (\sum_{i=1}^{n} \exp(\beta_i) - \exp(\beta_j)) \tag{110}$$

with $E(\text{avg}(\lambda_j)) \approx e^{\mu}$ and $\text{Var}(\text{avg}(\lambda_j)) \approx \exp(2 \cdot 0)(\Sigma_{11} + \Sigma_{22}) = \text{trace}(\Sigma)$. We remark that $E(f(X)) \neq f(E(X))$, so this is only an approximation, nor does it account for dependencies.

This value is not very interesting, as it doesn't help us determine the better team; all lamda-values are equal here. To get comparable lamdas, we instead look at $\lambda_j | y$. They don't have any nice looking expressions, so we instead use numeric methods to approximate the mean and the variance.

2.10.2 Model selection and fitting

Underfitting means to have a model that isn't sufficiently complex to model the target, i.e. the assumed model is too simple to accurately describe the target.

Overfitting is the opposite: having a model too complex for the target. This often tends to model the noise instead of the underlying distribution.

In both cases the fitted models fail to predict new data points. The aim in model selection is to find an optimal model that neither too strict or too flexible.

[16] Or rather minimize, as TMB works with the *negative* log-likelihood.

In figure 6 we have an example of a target distribution that is modelled by three different polynomials of different orders.

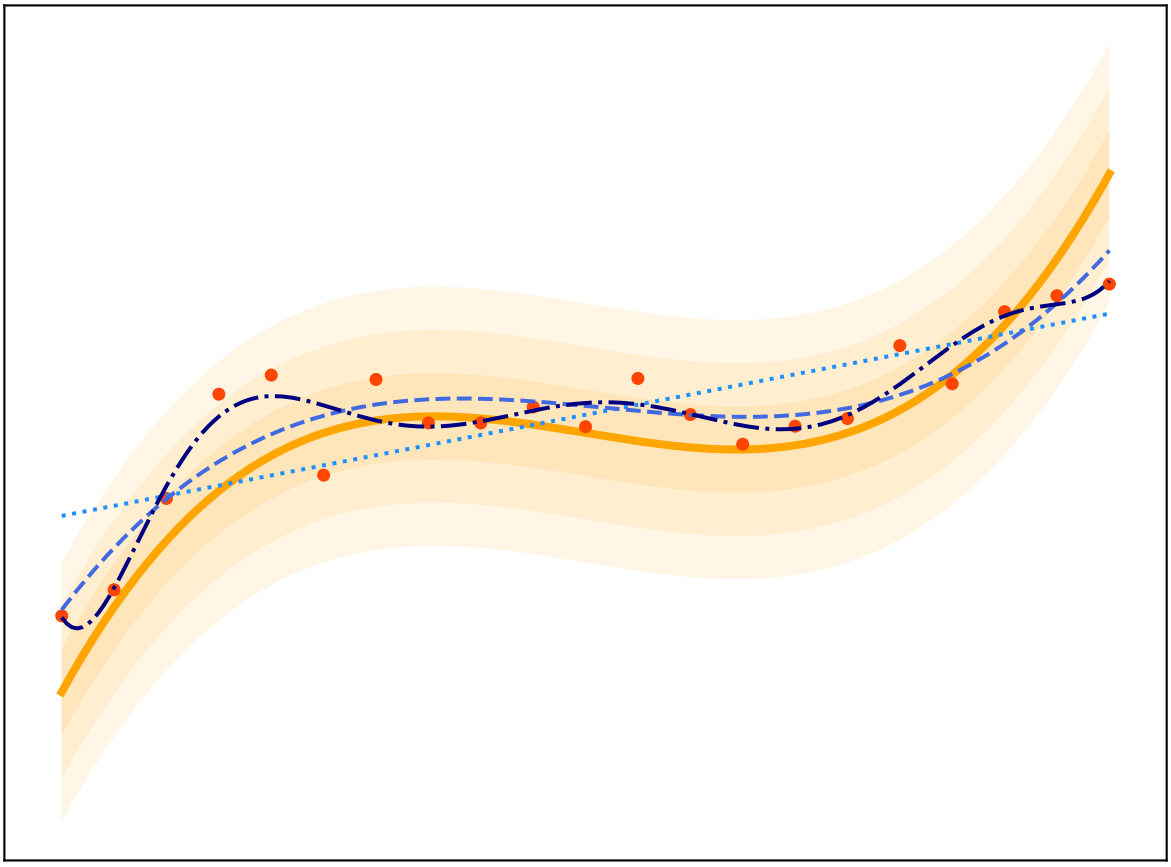

Figure 6: The orange solid line is the true underlying third order polynomial curve. The shaded orange area is 1, 2, and 3 standard deviations off the mean. The orange dots are the observations, or data, on which we perform polynomial regression. The light blue dotted line is an underfitted model, and is too simple. The blue dashed third order curve is a good model, and follows a similar trajectory to the true curve. The dark blue dash-dotted eighth order curve is an overfitted model, as it assumes a too complex model, and is biased towards obervations rather than the true curve.

Visually, we can see that the dashed blue model is "best". To find this mathematically we use information criterion values, such as AIC.

2.10.3 Akaike information criterion

AIC is a goodness-of-fit value, giving a lower value for better models. [2] A related value, which we call AIC star, is shown in the below equation:

$$\text{AIC}^* = \log(\hat{L}) - k \tag{111}$$

For small samples, one may subtract a correction term $\frac{k(k+1)}{n-k-1}$ to obtain the corrected criterion AICc*, but as this is approximately zero for large samples (assuming k is small), we may ignore this.

The theoretical derivation of AIC includes the quantity $-2\log(\hat{L})$, known as the *deviance*. So the usual definition of AIC is $\text{AIC} = -2 \cdot \text{AIC}^*$. Because the right hand side of the expression becomes easier to interpret, and it doesn't affect the ranking of the models, we omit the factor -2.

For the model in figure 6, we have information criterions for the three fitted models, and for five other polynomial models in figure 7.

Using the AICc*, we would correctly select the model with the same order as the target, but the coefficients would be different. The AIC would choose a fourth order curve, which is also close to the true order. Simply

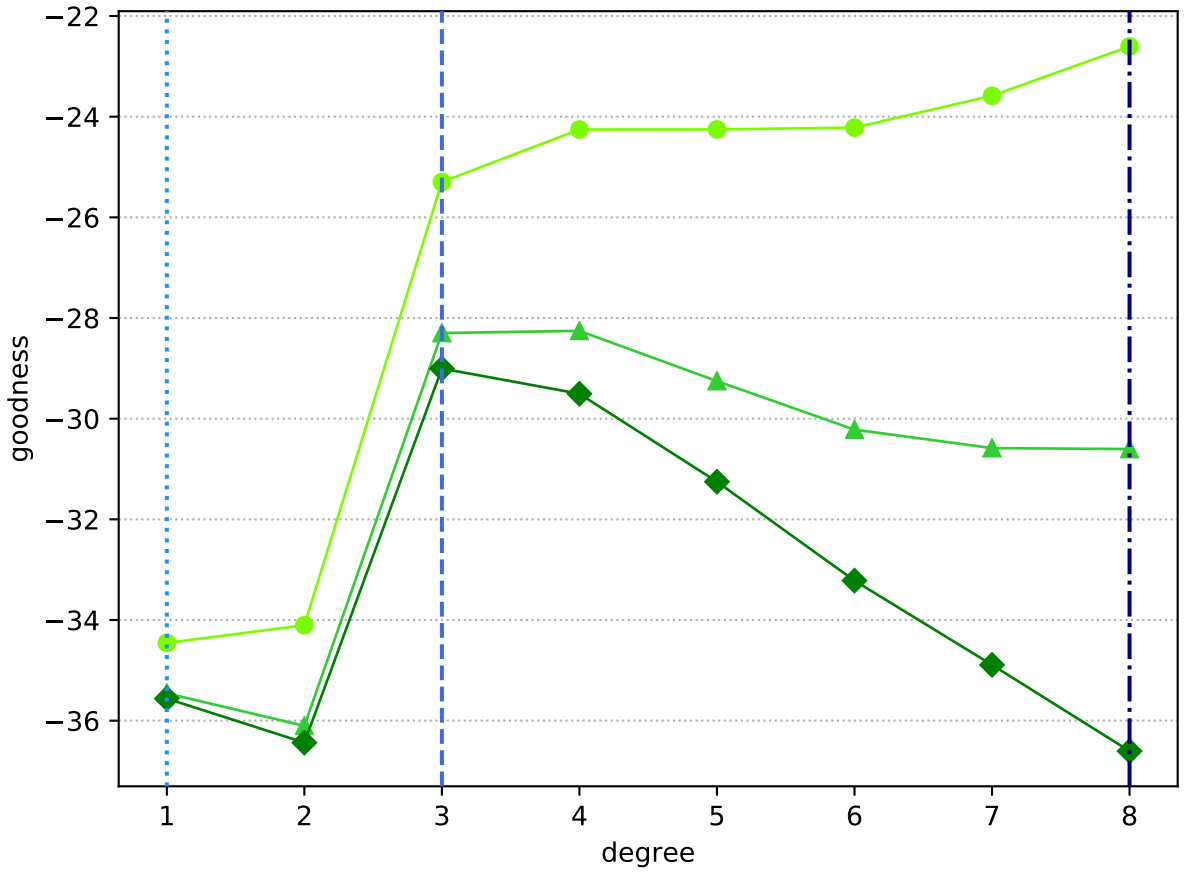

Figure 7: The values for $\log(\hat{L})$ (light green circles), AIC* (green triangles) and AICc* (dark green diamonds). They have a peak at 8, 4 and 3, respectively, in $[1, 8]$. The blue lines correspond to the fitted polynomials in figure 6 with the same pattern, indicating their fitness value.

relying on the likelihood would have selected an overfitted model; this is the reason for introducing penalizing terms.

The usefulness of AIC comes from its simple assumptions. It doesn't assume anything about the model; as long as we know the likelihood and the number of parameters, we can calculate the criterion value.

2.10.4 Numerical simulation and estimation of parameters

Using the normal posterior distribution for the attack and defence parameters, we can simulate new values by drawing from the distributions. This can be used to numerically estimate the mean and deviations of the λs, and the scores.

From repeated sampling of a season's result, we may obtain a distribution for a team's score for the given attack/defence parameters.

```
def season_result(A, gamma, mu):
    scores = np.zeros(shape=teams)

    result_point = {
        -1 : 0,
         0 : 1,
         1 : 3,
    }

    for home, away, hround, around in matches:
        hlamda = np.exp(A[hround,home,0] - A[around,away,1] + gamma + mu)
        alamda = np.exp(A[around,away,0] - A[hround,home,1] - gamma + mu)

        hscore = np.random.poisson(lam=hlamda)
```

```
        ascore = np.random.poisson(lam=alamda)

        result = np.sign(hscore - ascore)

        scores[home] += result_point[ result]
        scores[away] += result_point[-result]

    return scores
```

Listing 1: Example code using Python to simulate the result of a season.

2.11 Prediction

2.11.1 Naive prediction

While not unexpected, home teams usually have an advantage in matches, often referred to as the *home advantage*. We model this by the γ parameter in our models. Statistically, the home team wins roughly 45 % of the matches in football. [30] In comparison, the home team won 47 % of matches in Eliteserien 2019. [27], so without any knowledge about the teams, a good strategy would be to just bet on the home team. This will be correct almost half the time.

2.11.2 Most likely result outcome

The match result is either a win, tie or a loss.

Using the estimated values λ_i for each team in a given match, we can calculate the loss probability $P_L = P(k > 0|\lambda_{i1}, \lambda_{i2})$, the tie probability $P_T = P(k = 0|\lambda_{i1}, \lambda_{i2})$, and the win probability $P_W = P(k < 0|\lambda_{i1}, \lambda_{i2})$, using the skellam distribution. We then select the most likely result as our guess.

This method has a flaw in that P_T will always be smaller than either P_L or P_W, as long as $\lambda_{i1}, \lambda_{i2} \geq 1$ so it will never guess a tie. The proof is simple: $P_T(\lambda_{i1} = 1, \lambda_{i2} = 1) < \frac{1}{3}$, and is a maximum. I.e. increasing either λ_{i1} or λ_{i2} will make this value smaller. So either P_L or P_W must be greater than $\frac{1}{3}$, and be our guess.

However, around a third of matches result in a tie, [30], but figure 2 shows the average of λ_i to be around 1.5, so unless the model is really accurate and can estimates below 1, this method will be wrong approximately one third of the time.

2.11.3 Most likely score outcome

The match score is a pair of number, e.g. $1-4$, $2-2$ or $7-1$.

To make tie guesses more likely, we may want to use the mode instead. The mode represents the most likely score outcome, so instead of looking at what's most likely of a win, tie or loss, we look at each score outcome individually.

This makes sense because we fit the model to score, not the result of a match. However, it should more heavily favour ties, even when winning results combined would be more likely.

2.11.4 Weighted outcome

Another method that tries to correct the win bias, is to apply weights to the win, tie and loss probabilities. We then select the result with the highest weighted probability.

$$(n_W p_W, n_T p_T, n_L p_L) \tag{112}$$

$$n_W + n_T + n_L = 1 \tag{113}$$

$$(n_W, n_T, n_L) \geq 0 \tag{114}$$

3 Data analysis

3.1 Eliteserien 2019

The resulting scores from 2019 are taken from [27]. The scores are displayed in table 1.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1 Bodø/Glimt		1-2	2-2	3-0	4-0	0-0	3-2	3-0	5-1	2-0	1-1	3-3	2-0	4-0	2-1	4-0
2 Brann	1-1		0-0	2-1	1-0	0-0	0-0	1-0	0-1	0-1	2-1	2-1	1-1	2-3	1-5	1-1
3 Haugesund	1-1	1-1		0-0	0-2	0-0	1-2	4-1	0-1	2-1	1-1	3-0	2-2	5-1	1-0	1-4
4 Kristiansund	1-2	1-0	2-2		5-2	4-0	3-2	1-1	0-0	2-2	4-0	0-1	1-2	1-0	4-2	2-0
5 Lillestrøm	0-0	1-3	1-0	1-1		3-2	0-2	0-3	2-1	1-1	0-0	1-3	2-1	4-0	0-2	0-0
6 Mjøndalen	4-5	2-1	1-4	1-1	2-2		1-3	2-0	3-1	1-2	0-0	1-0	1-1	1-1	1-1	1-0
7 Molde	4-2	1-1	3-1	2-0	2-1	1-0		2-2	2-0	3-0	2-1	3-0	4-0	3-0	5-1	4-1
8 Odd	3-1	3-2	3-1	2-0	2-1	3-2	2-2		1-0	1-1	3-0	2-1	2-1	2-1	1-0	1-1
9 Ranheim TF	1-1	0-3	0-2	1-2	2-1	1-1	2-3	4-1		2-3	0-2	0-2	1-0	1-2	5-2	1-5
10 Rosenborg	3-2	0-0	0-2	1-0	3-1	3-2	3-1	1-1	3-2		1-0	3-2	0-0	5-2	5-1	3-0
11 Sarpsborg 08	1-1	1-1	1-1	0-1	1-0	1-1	1-1	2-0	1-3	1-1		0-0	2-2	3-2	2-2	1-0
12 Stabæk	2-0	0-1	1-1	2-0	1-1	4-2	1-2	0-0	0-0	3-1	3-3		2-1	0-1	0-0	1-1
13 Strømsgodset	1-3	6-0	3-2	2-3	1-1	2-3	0-4	2-3	1-0	3-3	2-1	0-2		3-1	0-0	3-2
14 Tromsø	1-2	1-2	2-2	5-0	1-1	2-2	2-1	1-2	4-2	1-0	2-0	1-1	0-1		0-2	0-0
15 Viking	3-4	2-1	0-0	2-0	3-0	4-1	0-2	2-0	2-2	2-2	2-1	3-0	4-0	2-1		1-1
16 Vålerenga	6-0	1-0	1-2	1-1	0-3	2-0	2-4	1-0	1-1	1-1	1-1	0-2	2-0	4-1	0-4	

Table 1: The result from 2019. The left number represents the home team score (row), the right number represents the away team score (column). The numbers in the column header correspond to the teams with the same number in the row header.

3.1.1 Data quirks

After using the data, a minor inconsistency was found: Rounds are not in order, so the number of games each team have played up until a match may differ. This is due to scheduling issues, so "round 2" may be moved to after "round 12", but round numbering is not renamed to account for this.[17]

4 Results

We first look at the estimated results for the rankings. These are summary statistics of the predictions of each match, which we will go through next.

4.1 Ranking

The estimated ranks are shown below. The first table is the time-independent model, shown in figure 8. The next three are the discrete models, shown in figure 9, figure 10 and figure 11. The last two are the continuous models, shown in figure 12 and figure 13.

We note that most estimated medians are drawn to around 45 points, away from the extremes.

4.1.1 Discrete models

4.1.2 Continuous models

4.1.3 Comparison of models

In table 2 we see the estimated parameter values for each model. In table 3 we have the AIC values for each model. By this measure, we see that the discrete random walk model to be the better one.

4.1.4 Ranking

Using the discrete random walk model, we can obtain the expected final standings, as shown in table 4. Only the top three teams were correctly predicted, however, most other scores weren't statistically different. With three teams with 40 points and four teams with 30 points, discrepencies ought to be expected.

4.2 Predicting the past

There are two ways to estimate results. One way is to use data from the entire season, and "predict the past". Or we can use all matches up to a certain date and predict the following match(es), and "predict the future". We first present the past predictions.

[17] I attribute this to my lack of football expertise, as this may be common knowledge among football fanatics.

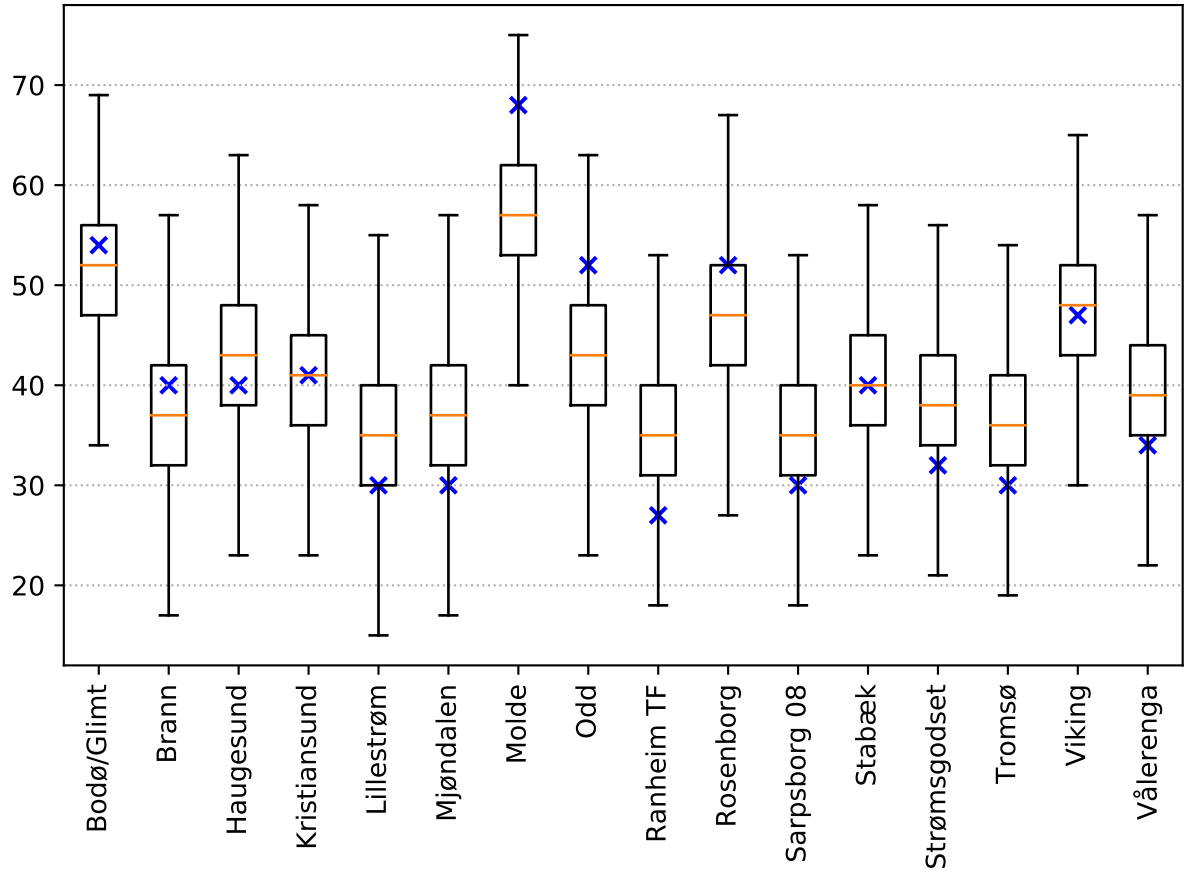

Figure 8: Box plot of the estimated points from 100K simulated seasons. The orange line is the median, and the blue cross is the actual points the team got in the season.

For these predictions we have used the continuous VAR model, for no particular reason; the discrete random walk model might have been a better choice.

For the most likely result rule, we can see the predicted result of each match in table 5. The table is very information dense, so we have the confusion matrixes below.

We see the confusion matrixes of the past predictions below in table 6, table 7 and table 8. For the weighted rule, we found the weights $(0.5, 0.3, 0.2)$, so this weighs loss probabilites more.

The weighted score outperformed the two other rules, but the weights are likely to be biased, and may not apply to other datasets.

4.3 Predicting the future

The future predictions involved fitting the model to all matches before a certain date, and make a prediction for the next match using the fitted model. So the model parameters change for each prediction.

As with the past predidction, we have the predicted result of each match in table 9 using the most likely result rule. Most of this table is uninteresting, but we note the first two matches between Odd-Brann and Vålerenga-Mjøndalen. Because we have no prior knowledge of their performance, we are unable to make predictions for these matches, so we only predict 238 matches.

While we are mostly interested in the proportion of correct guesses at the end of the season, it's also interesting to see how this evolves during the season, as shown in figure 14. It's relatively stable, but weaker than the past prediction methods.

We see the confusion matrixes of the future predictions below in table 10, table 11 and table 12. We use the same weights for the weighted rule as we did for past predictions, i.e. $(0.5, 0.3, 0.2)$; this introduces some bias. This can be remedied by updating these weights for each round as well.

We note that all the results of the future predictions are worse than the result of past predictions. This is expected, as we have less data to rely on.

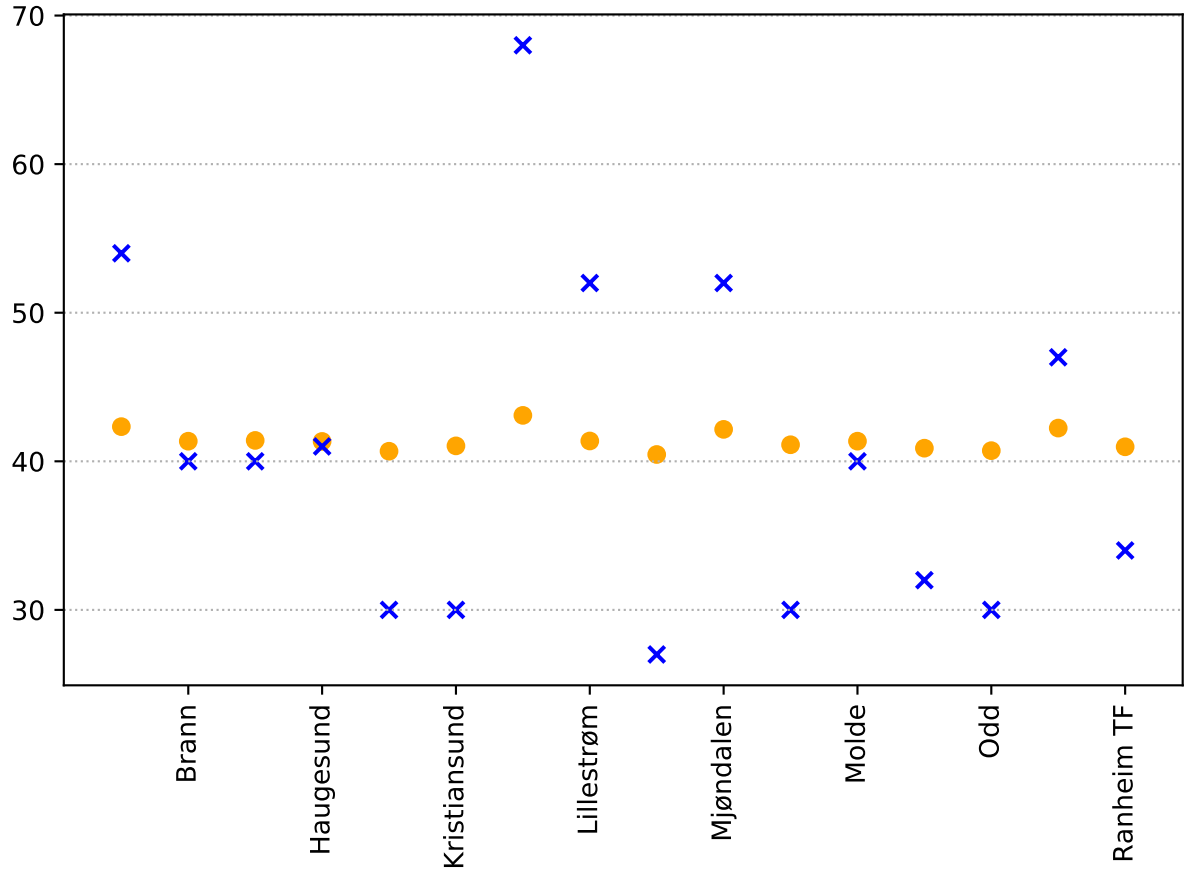

Figure 9: Box plot of the estimated points from 100K simulated seasons. The orange line is the median, and the blue cross is the actual points the team got in the season. Apparently this model were ill-defined, as the deviation was undefined.

5 Discussion & conclusion

We have seen that football matches are hard to predict, but that TMB is a useful tool in determining this. TMB made it easy to create and fit multiple models, without prior knowledge of dual numbers or laplace approximation, but simply through the likelihood function.

While the time series model had some issues, and most not better than a time independent model (as shown in table 3), they were interesting to study and model.

5.1 Further work

This paper has mostly been exploratory, and points to several directions that can be explored. We go through a few paths one may use for continued study.

We can change the values we fit the model to. A drawback of the points system in football is that wins are given extra weight with three points. So a single goal may add two points. When fitting, we tuned our parameters to the goal counts, not the result (win, tie, loss). One could instead use the result for fitting.

The model can be extended to include multiple seasons. This may improve the prediction precision, and get more accurate parameter estimates. Though, this may be difficult as some teams are removed and new ones added every season.

While predicting match and standing results is interesting, there are other results which would be interesting to predict, such the odds. These are usually also available from betting sites, and could be interesting to compare.

On the other end, it may be "obvious" to include more data to fit the model, such as player age, or match length, travel distance between matches, they may also be be redundant, and lead to overfitting.

Variables such as seasonal change were unaccounted for; while this falls under the same category as being prone to overfitting, it's possible for a team to progressively become better or worse during a season.

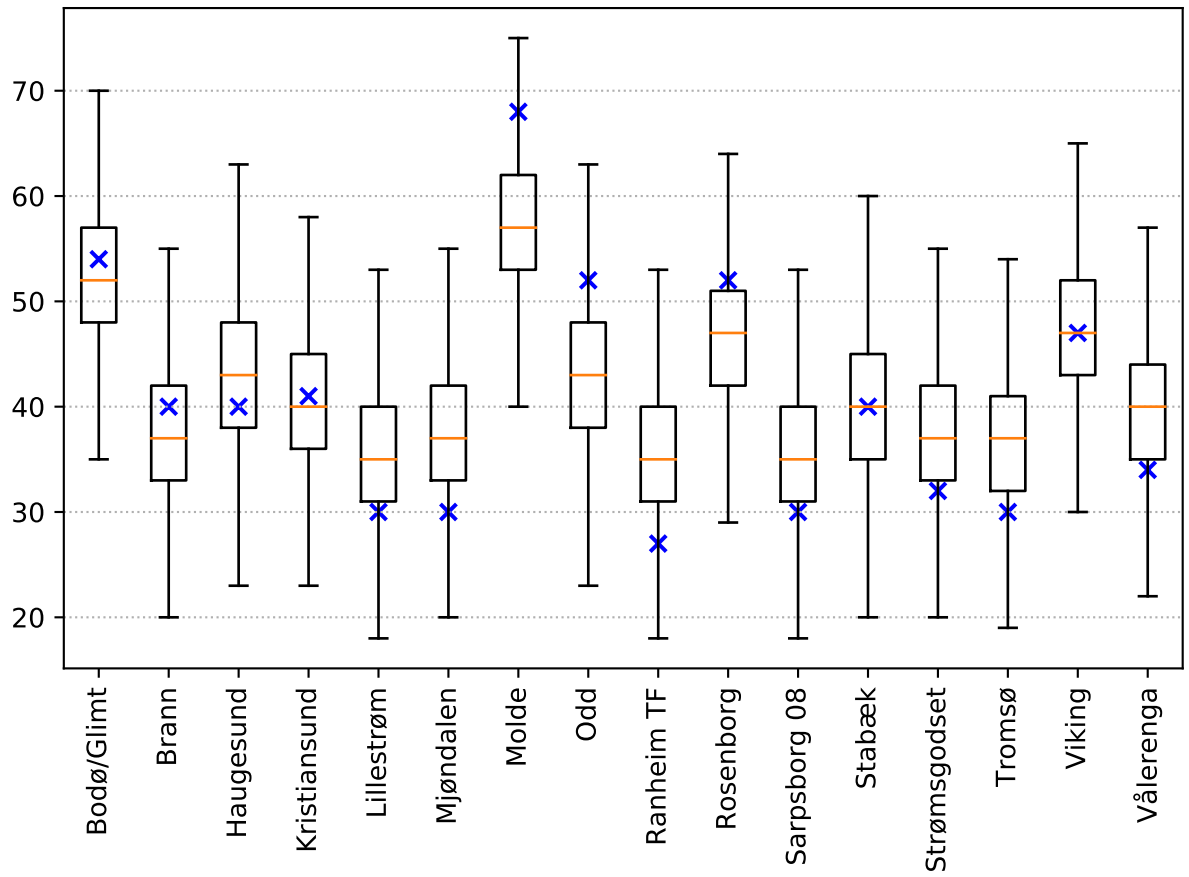

Figure 10: Box plot of the estimated points from 100K simulated seasons. The orange line is the median, and the blue cross is the actual points the team got in the season.

During the idea stage of the paper, the idea of intransitive ordering between teams came up. This would have been interesting to study further.

6 Appendix

6.1 Kronecker product and sum

The kronecker product and sum are useful for solving matrix equations. The product has a simple definition, but the sum is more complicated, and is related to the product by the matrix exponential.

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix} \tag{115}$$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \otimes \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} a\begin{pmatrix} e & f \\ g & h \end{pmatrix} & b\begin{pmatrix} e & f \\ g & h \end{pmatrix} \\ c\begin{pmatrix} e & f \\ g & h \end{pmatrix} & d\begin{pmatrix} e & f \\ g & h \end{pmatrix} \end{pmatrix} = \begin{pmatrix} ae & af & be & bf \\ ag & ah & bg & bh \\ ce & cf & de & df \\ cg & ch & dg & dh \end{pmatrix} \tag{116}$$

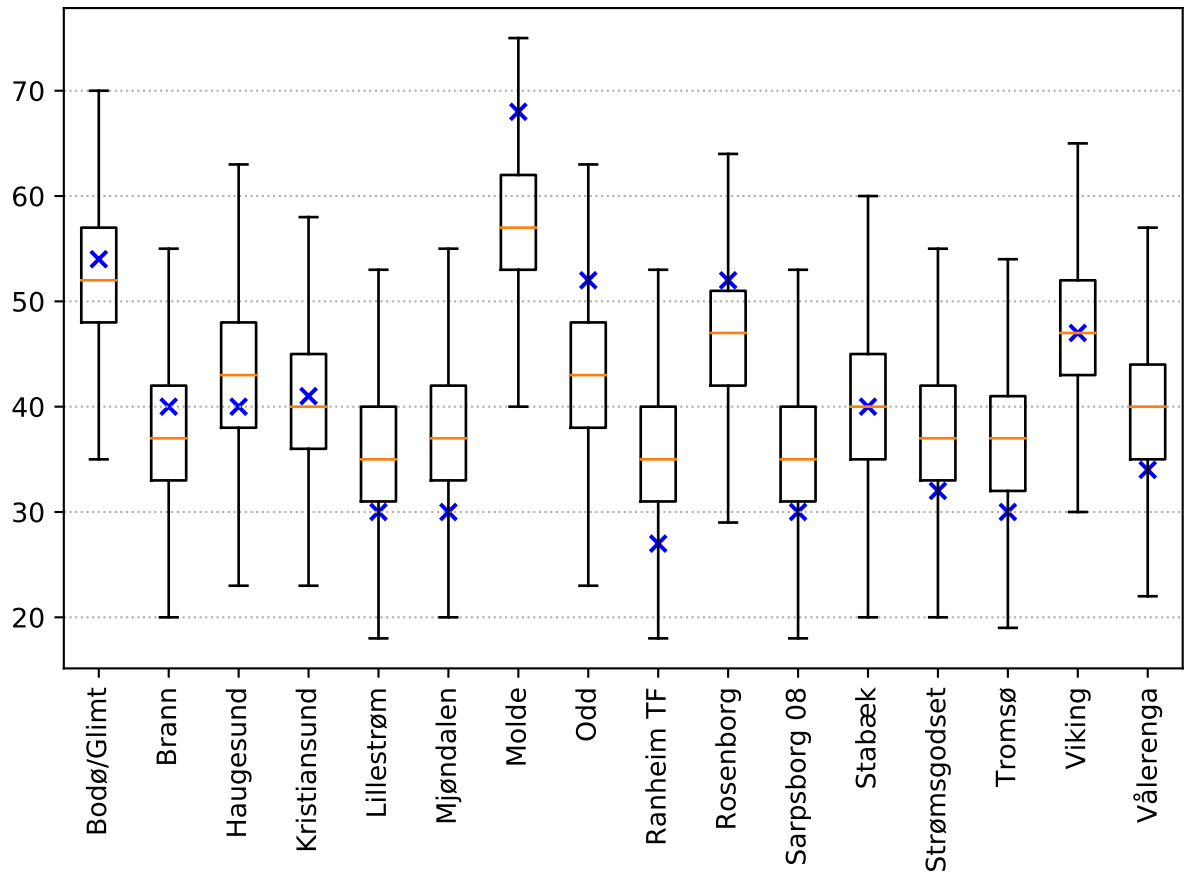

Figure 11: Box plot of the estimated points from 100K simulated seasons. The orange line is the median, and the blue cross is the actual points the team got in the season.

Before we define the sum, we will give two more examples to show that kronecker product is not commutative:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \otimes I = \begin{pmatrix} a & 0 & b & 0 \\ 0 & a & 0 & b \\ c & 0 & d & 0 \\ 0 & c & 0 & d \end{pmatrix} \tag{117}$$

$$I \otimes \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} e & f & 0 & 0 \\ g & h & 0 & 0 \\ 0 & 0 & e & f \\ 0 & 0 & g & h \end{pmatrix} \tag{118}$$

While the left hand sides are different in the two equations, the structure of the resulting matrix is clearly different.

The kronecker sum is defined as $A \oplus B = A \otimes I + I \otimes B$.

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \oplus \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} a & 0 & b & 0 \\ 0 & a & 0 & b \\ c & 0 & d & 0 \\ 0 & c & 0 & d \end{pmatrix} + \begin{pmatrix} e & f & 0 & 0 \\ g & h & 0 & 0 \\ 0 & 0 & e & f \\ 0 & 0 & g & h \end{pmatrix} = \begin{pmatrix} a+e & f & b & 0 \\ g & a+h & 0 & b \\ c & 0 & d+e & f \\ 0 & c & g & d+h \end{pmatrix} \tag{119}$$

This is notably not commutative either, which is unexpected for something called a sum.

The relation to the matrix exponential is given by $\exp(A) \otimes \exp(B) = \exp(A \oplus B)$.

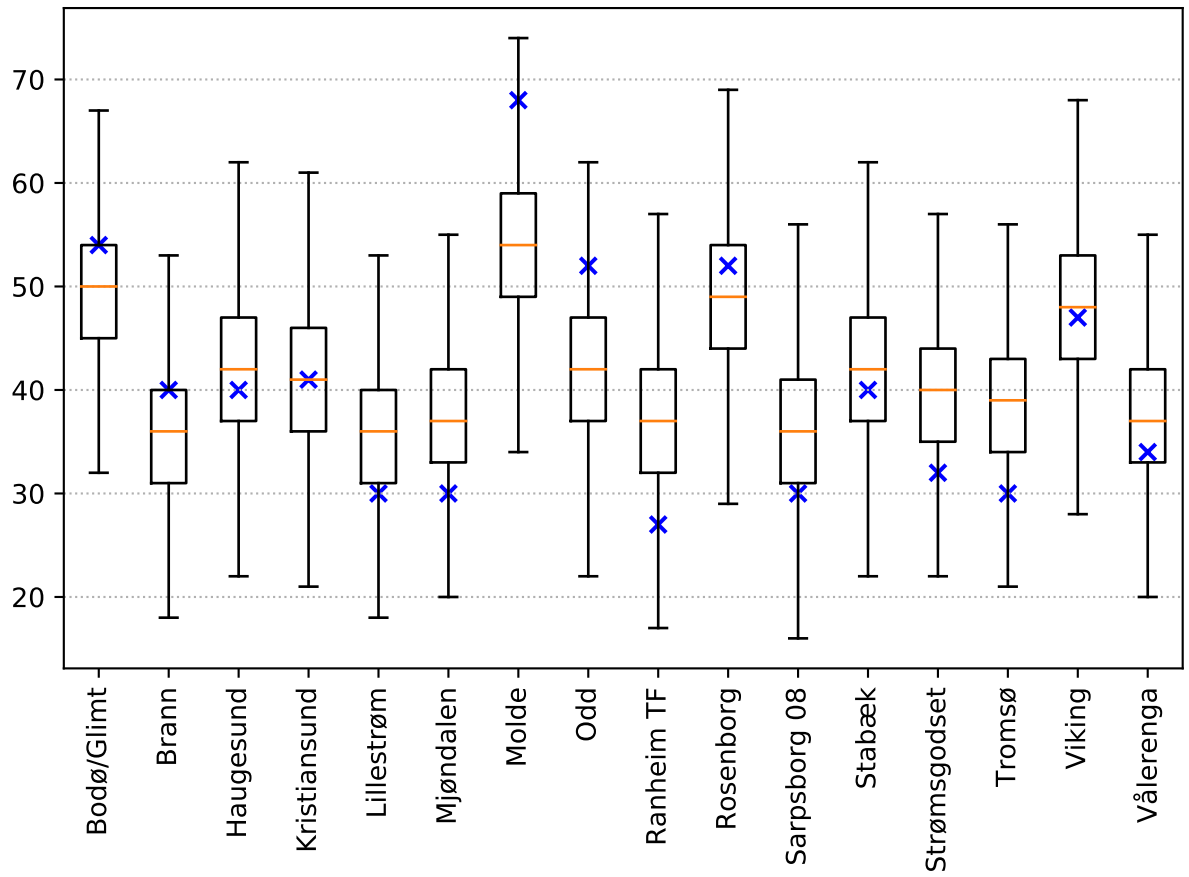

Figure 12: Box plot of the estimated points from 100K simulated seasons. The orange line is the median, and the blue cross is the actual points the team got in the season.

6.2 Solving the continuous VAR(1) differential

$$\mathrm{d}(x_t e^{\theta t}) = \theta x_t e^{\theta t}\mathrm{d}t + e^{\theta t}\mathrm{d}x_t \tag{120}$$

$$= \theta x_t e^{\theta t}\mathrm{d}t + e^{\theta t}(\theta(\mu - x_t)\mathrm{d}t + \sigma \mathrm{d}W_t) \tag{121}$$

$$= \cancel{\theta x_t e^{\theta t}\mathrm{d}t} + \theta\mu e^{\theta t}\mathrm{d}t - \cancel{\theta x_t e^{\theta t}\mathrm{d}t} + \sigma e^{\theta t}\mathrm{d}W_t \tag{122}$$

$$= \theta\mu e^{\theta t}\mathrm{d}t + \sigma e^{\theta t}\mathrm{d}W_t \tag{123}$$

$$\int_{s=0}^{t} \mathrm{d}(x_s e^{\theta s}) = \int_{s=0}^{t} (\theta\mu e^{\theta s}\mathrm{d}s + \sigma e^{\theta s}\mathrm{d}W_s) \tag{124}$$

$$= \int_{s=0}^{t} \theta\mu e^{\theta s}\mathrm{d}s + \int_{s=0}^{t} \sigma e^{\theta s}\mathrm{d}W_s \tag{125}$$

$$= \mu\int_{s=0}^{t} \mathrm{d}(e^{\theta s}) + \sigma\int_{s=0}^{t} e^{\theta s}\mathrm{d}W_s \tag{126}$$

$$[x_s e^{\theta s}]_{s=0}^{t} = \mu[e^{\theta s}]_{s=0}^{t} + \sigma\int_{s=0}^{t} e^{\theta s}\mathrm{d}W_s \tag{127}$$

$$x_t e^{\theta t} - x_0 = \mu(e^{\theta t} - 1) + \sigma\int_{s=0}^{t} e^{\theta s}\mathrm{d}W_s \tag{128}$$

$$x_t e^{\theta t} = x_0 + \mu(e^{\theta t} - 1) + \sigma\int_{s=0}^{t} e^{\theta s}\mathrm{d}W_s \tag{129}$$

$$x_t = x_0 e^{-\theta t} + \mu(1 - e^{-\theta t}) + \sigma\int_{s=0}^{t} e^{-\theta(t-s)}\mathrm{d}W_s \tag{130}$$

6.3 TMB workflow

```
# Data values
```

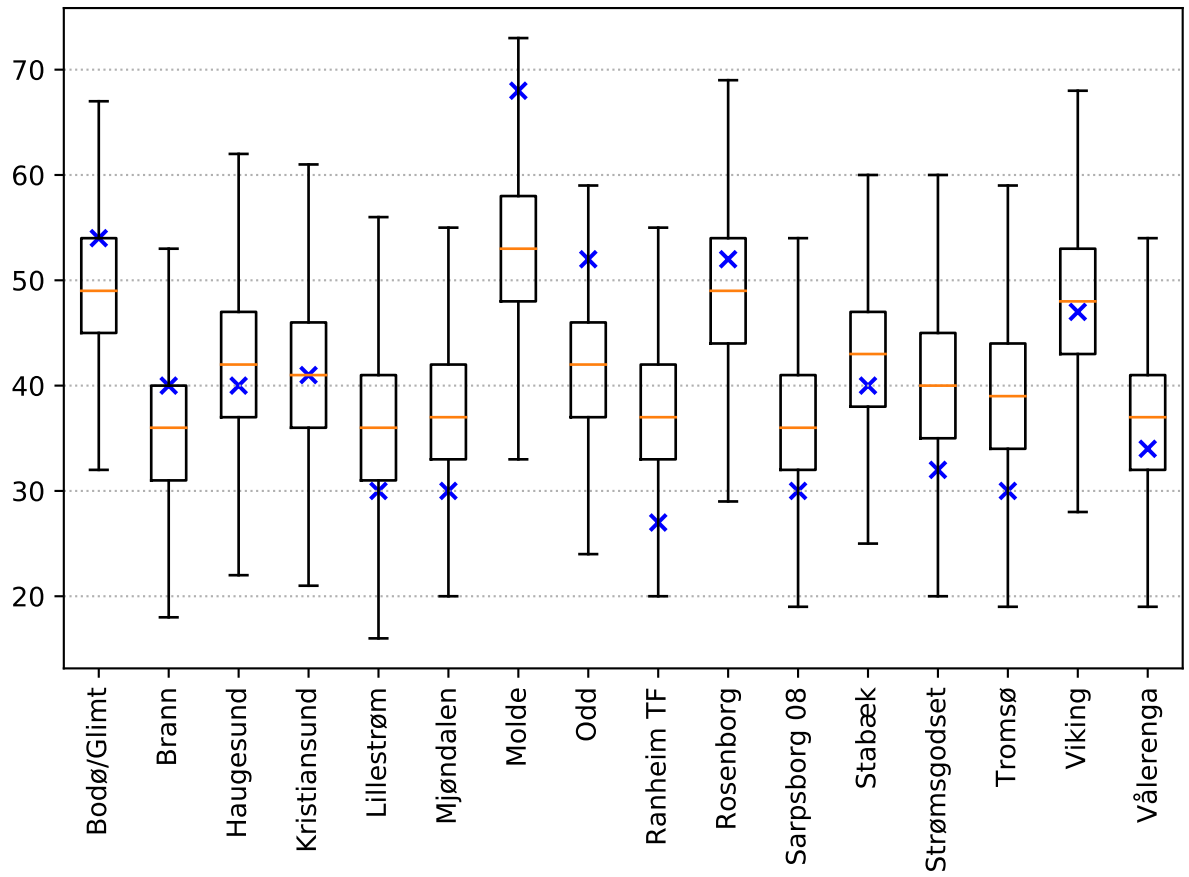

Figure 13: Box plot of the estimated points from 100K simulated seasons. The orange line is the median, and the blue cross is the actual points the team got in the season.

```
data <- list(...)

# Random & fixed effects
parameters <- list(...)


# Objective function
MODEL = "discrete_rw"

# Compile and link the template
. <- TMB::compile(paste0("models-tmb/", MODEL, ".cpp"))
dyn.load(TMB::dynlib(paste0("models-tmb/", MODEL)))


# Make Automatic Differentiation Function
obj <- TMB::MakeADFun(data, parameters, random=c("A"), DLL=MODEL, silent=TRUE)

# NonLinear MINimization subject to Box constraints
system.time(opt <- nlminb(obj$par, obj$fn, obj$gr))


# Result summary
report <- TMB::sdreport(obj)
```

Listing 2: Excerpt of a TMB program. The ellipsis would be replaced with the model values and parameters. A complete example may be found in the TMB paper , its documentation . For the program used in this thesis, we refer to its github repository, .

	Model	Parameter	Estimate	Standard error
T.I.	Noise	μ	$3.349e-01$	$5.138e-02$
		γ	$1.926e-01$	$3.852e-02$
		Σ_w	$\begin{pmatrix} 3.759e-02 & 1.313e-02 \\ 1.313e-02 & 5.694e-03 \end{pmatrix}$	$\begin{pmatrix} 2.067e-02 & 1.138e-02 \\ 1.138e-02 & 1.031e-02 \end{pmatrix}$
	Model	Parameter	Estimate	Standard error
Discrete	White noise	μ	$3.181e-01$	$5.348e-02$
		γ	$1.927e-01$	$3.863e-02$
		ϕ	0	0
		Σ_w	$\begin{pmatrix} 2.083e-02 & 1.709e-03 \\ 1.709e-03 & 2.988e-02 \end{pmatrix}$	$\begin{pmatrix} \text{NaN} & 2.446e-02 \\ 2.446e-02 & \text{NaN} \end{pmatrix}$
	Vector autoregressive	μ	$3.285e-01$	$5.393e-02$
		γ	$1.925e-01$	$3.857e-02$
		ϕ	$\begin{pmatrix} 1.025e+00 & 5.838e-02 \\ -1.401e-01 & 8.442e-01 \end{pmatrix}$	$\begin{pmatrix} 1.058e-01 & 9.009e-02 \\ 2.460e-01 & 2.085e-01 \end{pmatrix}$
		Σ_w	$\begin{pmatrix} 1.274e-03 & -4.291e-04 \\ -4.291e-04 & 1.445e-04 \end{pmatrix}$	$\begin{pmatrix} 2.802e-03 & 8.156e-04 \\ 8.156e-04 & 5.687e-04 \end{pmatrix}$
	Random walk	μ	$3.181e-01$	$5.319e-02$
		γ	$1.927e-01$	$3.863e-02$
		ϕ	1	0
		Σ_w	$\begin{pmatrix} 3.484e-03 & 8.236e-04 \\ 8.236e-04 & 1.947e-04 \end{pmatrix}$	$\begin{pmatrix} 1.780e-03 & 4.143e-04 \\ 4.143e-04 & 9.643e-05 \end{pmatrix}$
	Model	Parameter	Estimate	Standard error
Continuous	Vector autoregressive	μ	$3.181e-01$	$5.348e-02$
		γ	$1.927e-01$	$3.863e-02$
		θ	$\begin{pmatrix} 2.302e-06 & -9.381e-06 \\ -9.381e-06 & 3.975e-05 \end{pmatrix}$	$\begin{pmatrix} 1.043e-03 & 4.286e-03 \\ 4.286e-03 & 1.815e-02 \end{pmatrix}$
		D	$\begin{pmatrix} 5.889e-10 & 1.274e-10 \\ 1.274e-10 & 3.029e-11 \end{pmatrix}$	$\begin{pmatrix} 2.380e-07 & 5.144e-08 \\ 5.144e-08 & 1.234e-08 \end{pmatrix}$
	Random walk	μ	$3.215e-01$	$5.134e-02$
		γ	$1.923e-01$	$3.863e-02$
		θ	0	0
		Σ_w	$\begin{pmatrix} 4.459e-04 & 1.081e-04 \\ 1.081e-04 & 2.620e-05 \end{pmatrix}$	$\begin{pmatrix} 2.289e-04 & 5.465e-05 \\ 5.465e-05 & 1.230e-05 \end{pmatrix}$

Table 2: T.I. is the time independent model. These are the estimated parameters for each model. We have omitted the hundreds of values of α and β, but they were of similar order of magnitude as μ and γ. While μ and γ are significantly different from 0, the same cannot be said for the parameters of the time series.

	Model	AIC
T.I.	Noise	1462.915
	Model	AIC
Discrete	White noise	1476.872
	Vector autoregressive	1469.526
	Random walk	1461.973
	Model	AIC
Cont.	Vector autoregressive	1467.973
	Random walk	1462.855

Table 3: T.I. is the time independent model. Cont. are the continuous models. The best model, according to the calculated values of AIC, is discrete random walk model.

Rank	2019 result	TMB RW prediction
1	Molde (68)	Molde (54.707) v
2	Bodø/Glimt (54)	Bodø/Glimt (49.680) v
3	Rosenborg (52)	Rosenborg (48.989) v
4	Odd (52)	Viking (47.537) x
5	Viking (47)	Stabæk (42.080) x
6	Kristiansund (41)	Haugesund (41.905) x
7	Haugesund (40)	Odd (41.779) x
8	Stabæk (40)	Kristiansund (40.890) x
9	Brann (40)	Strømsgodset (39.674) x
10	Vålerenga (34)	Tromsø (38.654) x
11	Strømsgodset (32)	Mjøndalen (37.138) x
12	Sarpsborg 08 (30)	Vålerenga (37.105) x
13	Mjøndalen (30)	Ranheim TF (36.674) x
14	Lillestrøm (30)	Sarpsborg 08 (35.836) x
15	Tromsø (30)	Lillestrøm (35.748) x
16	Ranheim TF (27)	Brann (35.538) x

Table 4: Because most teams almost got the same number of points, they wouldn't be statistically different. There are also special rules for deciding the ranking of tied point scores, but we ignore this as we use real-valued points, truncated to three decimals.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1 Bodø/Glimt		H x	H x	H v	H v	H x	H v	H v	H v	H v	H x	H x	H v	H v	H v	H v
2 Brann	A x		H x	H v	H v	H x	A x	H v	H x	H x	H v	H v	H x	H x	A v	H x
3 Haugesund	H x	H x		H x	H x	H x	H x	H v	H x	H v	H x	H v	H x	H v	H v	H x
4 Kristiansund	H x	H v	H x		H v	H v	A x	H x	H x	H x	H v	H x	H x	H v	H v	H v
5 Lillestrøm	A x	H x	H v	H x		H v	A v	H x	H v	H x	H x	H x	H v	H v	A v	H x
6 Mjøndalen	H x	H v	H x	H x	H x		A v	H v	H v	A v	H x	H v	H x	H x	H x	H v
7 Molde	H v	H x	H v	H v	H v	H v		H x	H v	H v	H v	H v	H v	H v	H v	H v
8 Odd	H v	H v	H v	H v	H v	H v	A x		H v	H x	H v	H v	H v	H v	H v	H x
9 Ranheim TF	A x	H x	H x	H x	H v	H x	A v	H v		A v	H x	H x	H v	H x	H v	H x
10 Rosenborg	H v	H x	H x	H v	H v	H v	H v	H x	H v		H v	H v	H x	H v	H v	H v
11 Sarpsborg 08	A x	H x	H x	H x	H v	H x	A x	H v	H x	A x		H x	H x	H v	A x	H v
12 Stabæk	H v	H x	H x	H v	H x	H v	A v	H x	H x	H v	H x		H v	H x	H x	H x
13 Strømsgodset	A v	H v	H v	H x	H x	H x	A v	H x	H v	H x	H v	H x		H v	H x	H v
14 Tromsø	A v	H x	H x	H v	H x	H x	A x	H x	H v	H v	H v	H x	H x		H x	H x
15 Viking	H x	H v	H x	H v	H v	H v	H x	H v	H x	H x	H v	H v	H v	H v		H x
16 Vålerenga	A x	H v	H x	H x	H x	H v	A v	H v	H x	A x	H x	H x	H v	H v	A v	

Table 5: The predicted result given by the most likely result rule. Correct prediction are marked with a green cell (same shade as the top-right corner) on the web or with a `v` in the printed version. Incorrect predictions are orange or marked with an `x`.

		Predicted		
		Away	Tie	Home
Actual	Away	13	0	41
	Tie	10	0	63
	Home	3	0	110

Table 6: The confusion matrix using the entire dataset to to model and predict the matches using the most likely *result* rule. (e.g. win or lose.) The diagonal are the correct guesses; around 51 % were correct.

		Predicted		
		Away	Tie	Home
Actual	Away	1	46	7
	Tie	1	49	23
	Home	0	50	63

Table 7: The confusion matrix using the entire dataset to to model and predict the matches using the most likely *score* rule. (e.g. 1-1 or 3-2.) The diagonal are the correct guesses; around 47 % were correct.

		Predicted		
		Away	Tie	Home
Actual	Away	30	0	24
	Tie	24	0	49
	Home	17	0	96

Table 8: The confusion matrix using the entire dataset to to model and predict the matches using a *weighted result* rule. The diagonal are the correct guesses; 52 % of the total were correct.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1 Bodø/Glimt		H x	H x	H v	H v	H x	H v	H v	H v	H v	H x	H x	H v	H v	H v	H v
2 Brann	H x		H x	H v	H v	H x	A x	H v	H x	H x	H v	H v	H x	H x	A v	H x
3 Haugesund	H x	H x		H x	H x	H x	H x	H v	H x	A x	H x	H v	H x	H v	H v	H x
4 Kristiansund	H x	H v	H x		A x	H v	H v	H x	H x	H x	H v	H x	H x	H v	H v	H v
5 Lillestrøm	A x	H x	H v	H x		H v	H x	H x	H v	H x	H x	H x	H v	H v	H x	H x
6 Mjøndalen	H x	H v	H x	H x	H x		A v	H v	H v	A v	H x	H v	H x	H x	H x	H v
7 Molde	H v	H x	H v	H v	H v	H v		H x	H v	H v	H v	H v	H v	H v	H v	H v
8 Odd	H v	NA	H v	H v	H v	H v	H x		H v	H x	H v	H v	H v	H v	H v	H x
9 Ranheim TF	A x	H x	H x	H x	H v	H x	H x	A x		H x	H x	H x	H v	H x	H v	H x
10 Rosenborg	H v	H x	H x	H v	H v	H v	H v	H x	H v		H v	H v	H x	H v	H v	H v
11 Sarpsborg 08	H x	H x	H x	H x	H v	H x	H x	H v	H x	H x		H x	H x	H v	H x	H v
12 Stabæk	H v	H x	H x	H v	H x	H v	A v	H x	H x	H v	H x		H v	H x	A x	H x
13 Strømsgodset	H x	H v	H v	H x	H x	H x	H x	A v	H v	H x	H v	H x		H v	H x	H v
14 Tromsø	H x	H x	H x	A x	H x	H x	H v	H x	H v	H v	H v	H x	H x		H x	H x
15 Viking	H x	H v	H x	H v	H v	H v	H x	H v	H x	H x	H v	H v	H v	H v		H x
16 Vålerenga	H v	H v	H x	H x	H x	NA	A v	H v	H x	H x	H x	H x	H v	H v	H x	

Table 9: The predicted result given by the most likely result rule. Correct prediction are marked with a green cell (same shade as the top-right corner) on the web or with a `v` in the printed version. Incorrect predictions are orange or marked with an `x`.

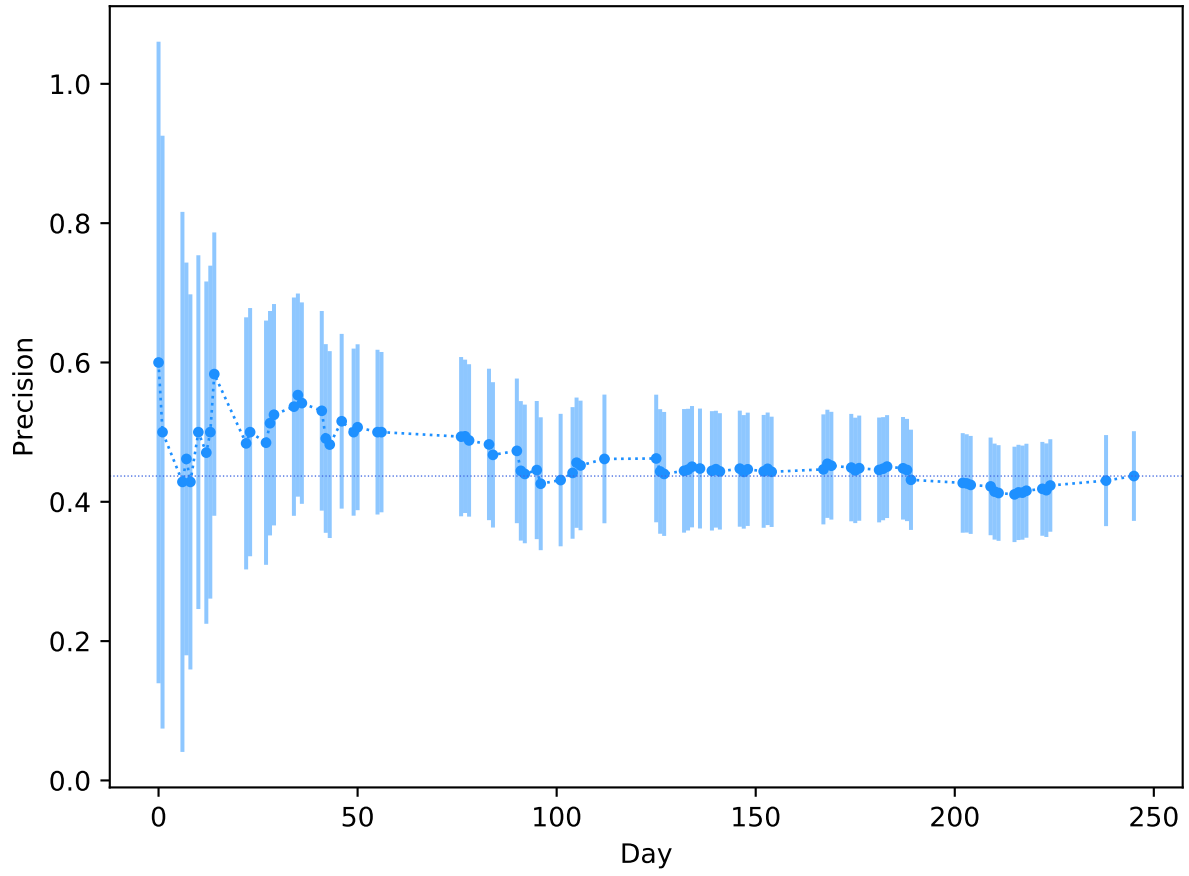

Figure 14: Precision over time. The normality assumption is violated near the first day, as the interval extends beyond 1, and p must be contained in $[0, 1]$. The confidence interval shrinks with more observations (matches). The intervals are four standard errors long.

		Predicted		
		Away	Tie	Home
Actual	Away	6	0	48
	Tie	4	0	69
	Home	4	0	107

Table 10: The confusion matrix using the entire dataset to to model and predict the matches using the most likely *result* rule. (e.g. win or lose.) The diagonal are the correct guesses; around 47 % were correct.

		Predicted		
		Away	Tie	Home
Actual	Away	0	41	13
	Tie	0	48	25
	Home	0	68	43

Table 11: The confusion matrix using the entire dataset to to model and predict the matches using the most likely *score* rule. (e.g. 1-1 or 3-2.) The diagonal are the correct guesses; around 38 % were correct.

		Predicted		
		Away	Tie	Home
Actual	Away	9	0	45
	Tie	15	0	58
	Home	16	0	95

Table 12: The confusion matrix using the entire dataset to to model and predict the matches using a *weighted result* rule. The diagonal are the correct guesses; 43 % of the total were correct.

References

[1] Abramowitz, M., Stegun, I. A. *Handbook of Mathematical Functions*. 8 2002.

[2] Akaike, H. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[3] Baio, G., Blangiardo, M. A. Bayesian hierarchical model for the prediction of football results. *Journal of Applied Statistics*, 37(2):253–264, 2010. [Online; accessed 2020-06-17].

[4] Basak, G. K., Lee, P. Asymptotic properties of an estimator of the drift coefficients of multidimensional Ornstein-Uhlenbeck processes that are not necessarily stable. *Electronic Journal of Statistics*, 9 2008.

[5] Brockwell, P. J., Davis, R. A. *Introduction to time series and forecasting*. 2 edition, 2010.

[6] Brugger, R. M. A Note on the Unbiased Estimation of the Standard Deviation. *The American Journal of Psychology*, 23(4):32, 1969.

[7] Carpenter, B., Hoffman, M. D., Brubaker, M., Lee, D., Li, P., Betancourt, M. The Stan Math Library: Reverse-Mode Automatic Differentiation in C++. *arXiv*, 9 2015. [Online; accessed 2020-07-12].

[8] Casella, G., Berger, R. L. *Statistical inference*. 2 edition, 2002.

[9] Ditlevsen, S. Overheads1b, 4 2008. [Online; accessed 2020-06-12].

[10] Finch, S. Ornstein–Uhlenbeck Process, 5 2004. [Online; accessed 2020-06-12].

[11] Holý, V., Tomanová, P. Estimation of Ornstein-Uhlenbeck Process Using Ultra-High-Frequency Data with Application to Intraday Pairs Trading Strategy, 12 2019. [Online; accessed 2020-06-12].

[12] Holtzman, W. H. The Unbiased Estimate of the Population Variance and Standard Deviation. *The American Journal of Psychology*, 63(4):615–617, 10 1950.

[13] Jacob, B., Guennebaud, G. and contributors. Eigen, 2020. [Online; accessed 2020-07-13].

[14] Jarle Tufto. Unconditional mean and variance of a stationary VAR(1) model, 12 2016. [Online; accessed 2020-04-29].

[15] Karlis, D., Ntzoufras, I. Analysis of sports data using bivariate Poisson models, 4 2003. [Online; accessed 2020-07-12].

[16] Karlis, D., Ntzoufras, I. Bayesian modelling of football outcomes: Using the Skellam's distribution for the goal difference., 8 2008. [Online; accessed 2020-07-23].

[17] kaskr and contributors. density::UNSTRUCTURED_CORR_t< scalartype_ > Class Template Reference, 2020. [Online; accessed 2020-06-22].

[18] kaskr and contributors. Template Model Builder (TMB), 2020. [Online; accessed 2020-06-15].

[19] Kass, R. E., Steffey, D. Approximate Bayesian Inference in Conditionally Independent Hierarchical Models (Parametric Empirical Bayes Models). *Journal of the American Statistical Society*, 84(407), 9 1989.

[20] Keng, B. Normal Approximation to the Posterior Distribution, 4 2016. [Online; accessed 2020-06-28].

[21] Kristensen, K. sdreport function, 2020. [Online; accessed 2020-07-12].

[22] Kristensen, K and Nielsen, K and Berg, C. W. and Skaug, H. and Bell, B. M. TMB: Automatic Differentiation and Laplace Approximation, 4 2016. [Online; accessed 2020-06-22].

[23] Kristensen, Kasper and Nielsen, Anders and Berg, Casper W. and Skaug, Hans and Bell, Brad. TMB: Automatic Differentiation and Laplace Approximation. *arXiv*, 2015.

[24] Langseth, H. Beating the bookie: A look at statistical models for prediction of football matches, 9 2013. [Online; accessed 2020-07-23].

[25] Luca Citi. How the Ornstein–Uhlenbeck process can be considered as the continuous-time analogue of the discrete-time AR(1) process?, 4 2017. [Online; accessed 2020-04-29].

[26] L.V.Rao. Derivation of change of variables of a probability density function?, 10 2016. [Online; accessed 2020-06-24].

[27] NIFS. football-data, 2019. [Online; accessed 2020-07-20].

[28] Oldřich Vašíček. An Equilibrium Characterization of the Term Structure. *Journal of Financial Economics*, 5(2):177–188, 1977.

[29] Ralph Allan Bradley and Milton E. Terry. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[30] Schoch, D. Home-field advantage, 6 2017. [Online; accessed 2020-07-20].

[31] Shewhart, W. A. Economic Control of Quality of Manufactured Product. *New York: D. Van Nostrand Company*, page 185, 1931.

[32] Skellam, J. G. The frequency distribution of the difference between two Poisson variates belonging to different populations. *Journal of the Royal Statistical Society*, 109(3):296, 1946.

[33] Tang, C. Parameter estimation and bias correction for diffusion processes and a nonparametric approach to census population size estimation, 2008. [Online; accessed 2020-06-12].

[34] Tokdar, S. T. Laplace Approximation to the Posterior, 2013. [Online; accessed 2020-06-28].

[35] Uhlenbeck, G. E. and Ornstein, L. S. On the Theory of the Brownian Motion. *Physical Review*, 36(5):823–841, 9 1930.

[36] Various. Binomial proportion confidence interval, 7 2020. [Online; accessed 2020-07-24].

[37] Vatiwutipong, P., Phewchean, N. Alternative way to derive the distribution of the multivariate Ornstein–Uhlenbeck process, 2019. [Online; accessed 2020-06-12].

[38] Wahl, J. C. Parameter Estimation of Multivariate Factor Stochastic Volatility Models. *Universitetet i Bergen*, 6 2018. [Online; accessed 2020-06-24].

NTNU

Norwegian University of
Science and Technology