fcfm
Ingeniería
FACULTAD DE CIENCIAS
FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

Tarea 1 - Nonlinear Regression

MA5204: Machine Learning 2019

Submitted To:
Felipe Tobar

Submitted By :
Alexandre Poupeau

1 Nonlinear Regression

1) Data analysis Here is the data divided into testing and training parts. The training part is composed of approximately 9 years :

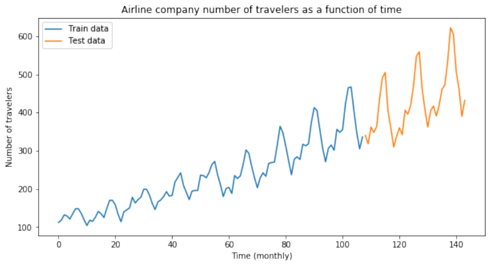

Figure 1: Graph of the data

What we can see is that there are fluctuations each year due to, I suppose, the different seasons (especially winter and summer). Overall, the number of passengers increases with time (we can suppose the company gets more famous and approved by people with the time) and the fluctuations also increases with the time.

2) Polynomial part We are going to try to find a model to fit the data using a polynomial base. Here is how we construct the model :

$y = f_\theta(x) + \eta = \Phi_d(x)\theta + \eta$ where $\Phi_d(x) = [1, x, x^2, ..., x^d]$ and d is the degree of the polynomial.

We suppose that η describes noise and therefore can be represented as following a Gaussian distribution centered in zero : $\eta \sim \mathcal{N}(0, \sigma_\eta^2)$.

Thus, we need to find the optimal $\theta' = [\theta, \sigma_\eta^2]$.

Finding the optimal θ' is equivalent to finding the θ' that maximize the following expression : $p(\theta'|data) \equiv p(data|\theta')p(\theta') = \frac{1}{\sqrt{2\pi\sigma_\eta^2}^N} \exp\left(\frac{-1}{2\sigma_\eta^2}\sum_{i=1}^{N}(y_i - \Phi_d(x_i)\theta)^2\right) * \frac{1}{\sqrt{2\pi\sigma_{\theta'}^2}} \exp\left(\frac{-1}{2\sigma_{\theta'}^2}\|\mu_{\theta'} - \theta'\|^2\right)$

We can suppose that all the information we have on the prior is $\theta' \sim \mathcal{N}(\mu_{\theta'}, \sigma_{\theta'}^2)$ where, to simplify everything, $\mu_{\theta'} = 0$ and $\sigma_{\theta'}^2 = 1$.

Thus $NLL(\theta') = -\log\left(p(\theta'|data)\right) \equiv \frac{N}{2}\log\left(\sigma_\eta^2\right) + \frac{N}{2\sigma_\eta^2} mean((Y - \Phi_d(X)\theta)^2) + \frac{1}{2}(\|\theta\|^2 + (\sigma_\eta^2)^2)$ where $\Phi_d(X) = [\Phi_d(x_1)^T, ..., \Phi_d(x_n)^T]^T$ and $Y = [y_1, ..., y_N]^T$.

We want to find $\theta'_{opt} = argmin(NLL(\theta'))$.

The previous formula does not provide the result θ'_{opt} in a closed-from. Hence we need to minimize the function using methods that aim to decrease the gradient such as Gradient Descent. However we can not obtain the result that easily and if we try with one initial condition it will not work. We have to imagine that we are in a ocean of possible initial conditions and that only some of them will converge to a local minimum or, with luck, the global minimum of $NLL(\theta')$. Therefore, I made a loop that creates random initial condition and only keeps the best model at the end (the one with the less test error). I made this process for the all the degrees from 1 to 4 of the polynomial. Here are the results :

Degree	Train Error	Test error
1	2309	4919
2	2277	7608
3	2621	4596
4	2180	4627

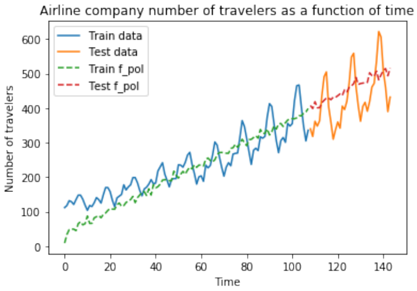

Figure 2: Prediction f_{pol} with degree 1

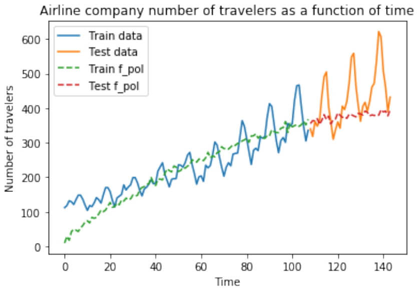

Figure 3: Prediction f_{pol} with degree 2

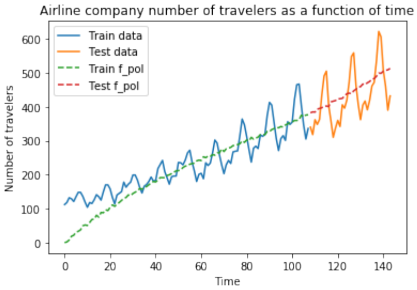

Figure 4: Prediction f_{pol} with degree 3

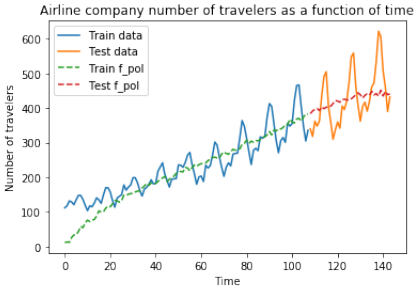

Figure 5: Prediction f_{pol} with degree 4

We can deduce that the model can not be a polynomial of degree 2 (at least of the form $f_{pol} = ax^2 + bx + c$ with $a < 0$ which is always obtained as optimal parameter) because the test error is too big. It is not intuitive because I would have thought that the degree would be 2 with a parameter $a > 0$ tiny (what I obtain using MCR, Minimo Cuadrado Regularizado see section 3). So what is the best degree then, when considering the noise ? It is hard to tell because all test error are really close, even thought it looks slightly worst for the lineal model. The degree 3 polynomial nonetheless has the worst train error and the best test error. I chose to think more about the problem conditions and context than about the test error to solve the degree question. In that case a degree 3 with $a > 0$ would mean that the number of travelers will increase a lot if we consider long times, which does not make sense in a commercial point of view. The number of travelers can not explode. Moreover, in the case of the degree 4 with $a < 0$, in a short time the number of travelers will drastically decrease which is not what we want either (except if the company expects a business failure). Finally my conclusion, even though it has to be approved by a commercial expert and discussed, would be the degree 1 given all the information we have so far. It is not the greatest model to describe the first part of the curve but it will make much more sense for future predictions.

3) First periodic component (FPC) In order to find the first periodic component, I used the same method as for the polynomial, no need to write the expression to minimize (and I do not have enough space !). Here is the result :

We can see that the model get the good period however it tries to minimize the error predicting a noise with a high variance which make the result looks kind of *unpredictable* in a way. The test error is around 2984 so it has decreased which is good. Thus it predicts better than before but this is not that good.

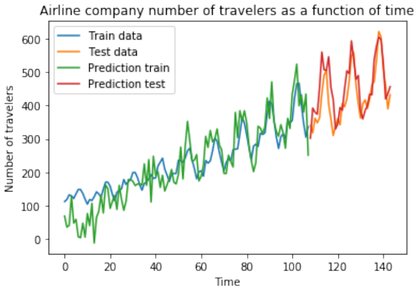

Figure 6: Prediction - 1st periodic component adding

4) Second periodic component (SPC) adding In order to find the second periodic component, I used the same method as for the first one. Here is the result :

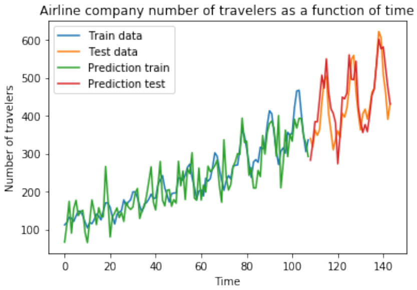

Figure 7: Prediction - 2nd periodic component adding

At the end the model has got a train error of 768 and a test error of 1453 approximately which is quite good objectively. Finally the polynomial part allows the model to get the idea of the main *direction* of the curve. The first periodic component allows the model to capture the main period present in the data and the amplitude that grows with the time. The second periodic component allows the model to capture more special fluctuations in the data. About the variance for each step, it allows the model to *correct* a bit the error by randomizing the prediction just a little bit around the mean of where should be precisely the expected point. This process allows the model to be realistic because we can not predict exactly the expected point, so it adds a little bit of randomness itself. It would have been possible in theory to create the final model with the $f_{pol} + FPC + SPC$ but it would have taken much more time to find good local minimums and we would have been obliged to iterate with more different initial conditions to explore more possibilities.

2 Project

For the project, I am going to work in a group of three on Recurrent Neural Networks applied to music style recognition to propose music suggestions based on what the listener is used to listen. The goal would be to classify musics into genres. At the end, we would like to be able to predict a music style. In our problem, the input is the music and the output the style. Music suggestions is quite easy once we have done that.

Link of the project context : `https://www.kaggle.com/c/mlp2016-7-msd-genre`

Link of the dataset we would use (the subset 1.9Gb / 10000 songs): `https://labrosa.ee.columbia.edu/millionsong/`.

3 More detailed (to read if you have time !)

Because THERE ARE SO MUCH MORE to be said on this problem, I made that part to go deeper. I found the best model using MCR first to get the optimal f_{pol} of degree of the form $f_{pol} = ax^2 + bx + c$ with $a > 0$ (tiny). Then using two periodic components without considering noise. It looks like adding noise makes the degree two polynomial going in the bad direction. What is interesting in this problem is that it does not seem to have a perfect indisputable answer like most real-life problems.

Here is the final best result I can provide for the initial problem.

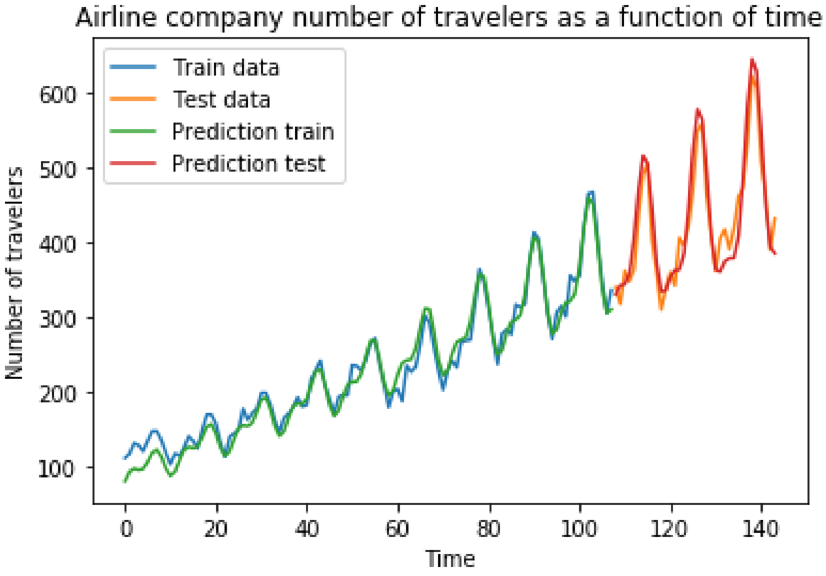

Figure 8: MCR for f_{pol} + FPC without noise + SPC without noise

The train error is 248, the test error is 841 which is clearly better than what we got using the previous method. However one could argue that this model is too perfect and he would be right. There is no noise at all.

I guess the model can get even better by adding other periodic components. The point of this part "More detailed (to read if you have time !)", is to question the use of the noise because in this particular problem it *does not seem* to improve the result one can get at the end.

Here is the extended tabular on the polynomial part in order to make a deeper analysis with other methods :

	Considering noise		Without considering noise		Using MCR method	
Degree	Train Error	Test Error	Train Error	Test Error	Train error	Test error
1	2309	4919	2204	5293	1215	4562
2	2277	7608	2155	8216	1110	4532
3	2621	4596	2584	4486	1108	6626
4	2180	4627	2091	4642	1108	4777

There is a lot to be said with all this information (but the report is already quite long...). If we do not consider noise, we find that the best model is with 3 degree and if we use the MCR methods we find 2 degree (and in that case the $a > 0$ with $f_{pol} = ax^2 + bx + c$).