Introduction to Machine Learning: Regression Module 1

Ehsan Shaghaei

Ehsan Shaghaei, Applied Artificial Intelligence Dept., Innopolis University
e.shaghaei@innopolis.university

1 Introduction

In this document, different regression models are investigated to fill the missing values in each features with a regression model for each of them. Furthermore, a data-set of email spam is used to train by logistic regression to find junk emails, and finally, a conclusion is made about the features that influenced the decisions of the model the most.

2 Linear/Polynomial Regression

2.1 Practical Task 1

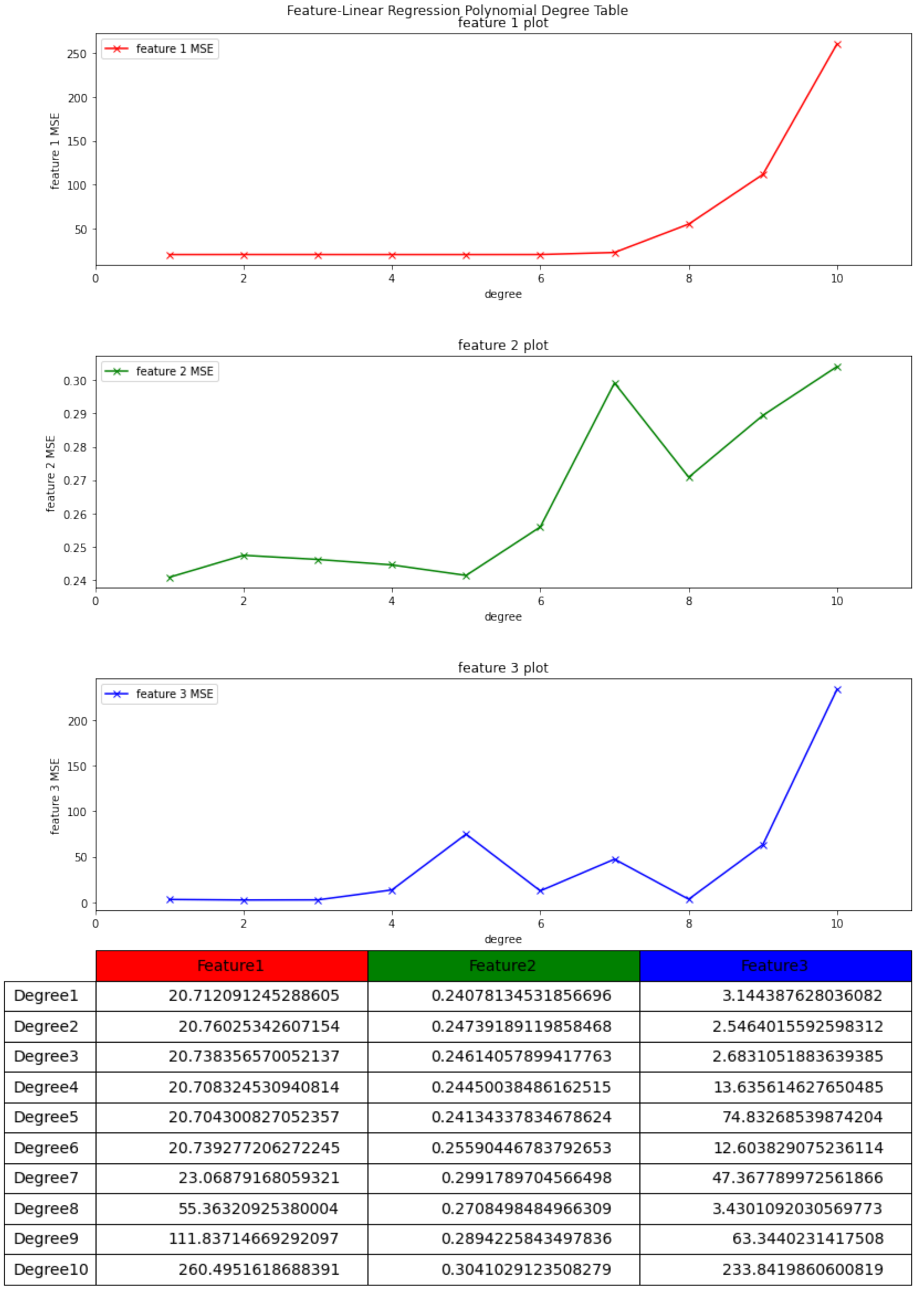
Regarding the comparison of obtained *"mean squared error"* for different degrees of polynomial regression in Fig. 1, conclusions are as following; the optimised degree for prediction of features 1,2 and 3 are degrees 5,1 and 2 respectively which results in $MSE_{f1}^{deg=5} = 20.704300827052357$, $MSE_{f2}^{deg=1} = 0.24078134531856696$ and $MSE_{f3}^{deg=2} = 2.5464015592598312$ because they imply the minimum *"mean squared error"* in comparison to other models in each feature.

2.2 Ridge Regression

In Ridge Regression, we assume that our system has a linear behavior. Moreover, Ridge regression similarly to simple linear regression tries to minimize the sum of the squared residuals; besides it also tries to decrease the variance of the model by penalizing the model parameters of the predicted linear model as the difference between Ridge regression and linear regression, however this penalization is highly affected by the scale of the features which is supposed to be standardised.

Now, let's analyse the value of the hyper-parameter λ in penalizing the θ as the parameter of the linear model. In Ridge regression we obtain parameters as following:

$$(\hat{\theta}, \hat{\theta_0}) = \underset{\theta,\theta_0}{\operatorname{argmin}}(\sum_{t=1}^{n}(y_t - \theta x_t - \theta_0)^2 + \lambda\theta^2)$$



	Feature1	Feature2	Feature3
Degree1	20.712091245288605	0.24078134531856696	3.144387628036082
Degree2	20.76025342607154	0.24739189119858468	2.5464015592598312
Degree3	20.738356570052137	0.24614057899417763	2.6831051883639385
Degree4	20.708324530940814	0.24450038486162515	13.635614627650485
Degree5	20.704300827052357	0.24134337834678624	74.83268539874204
Degree6	20.739277206272245	0.25590446783792653	12.603829075236114
Degree7	23.06879168059321	0.2991789704566498	47.367789972561866
Degree8	55.36320925380004	0.2708498484966309	3.4301092030569773
Degree9	111.83714669292097	0.2894225843497836	63.3440231417508
Degree10	260.4951618688391	0.3041029123508279	233.8419860600819

Fig. 1. Task 2.1

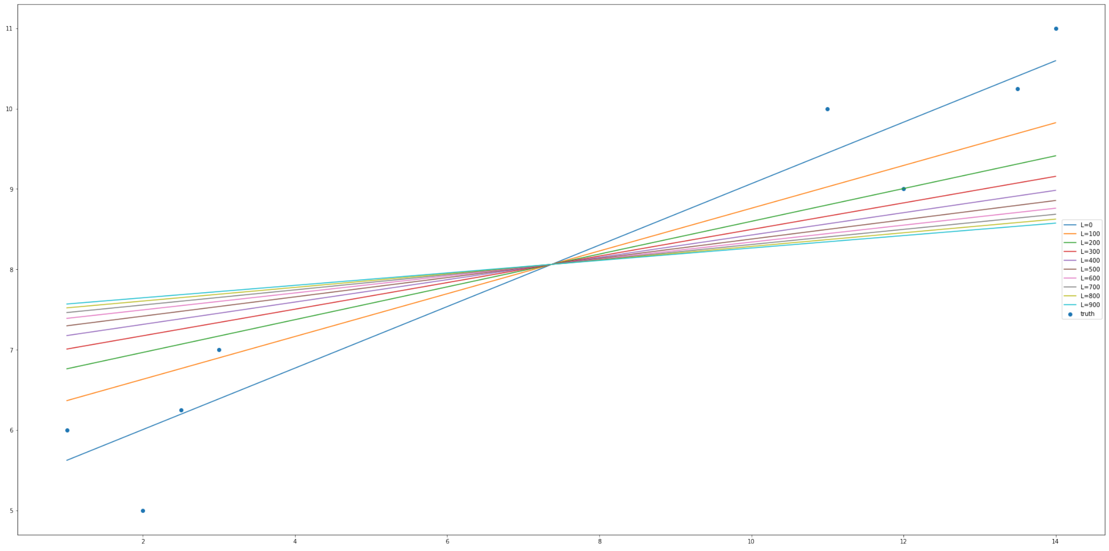


Fig. 2. Task 2.2-Examining different range of $0 \geq \lambda$ for Ridge Regression

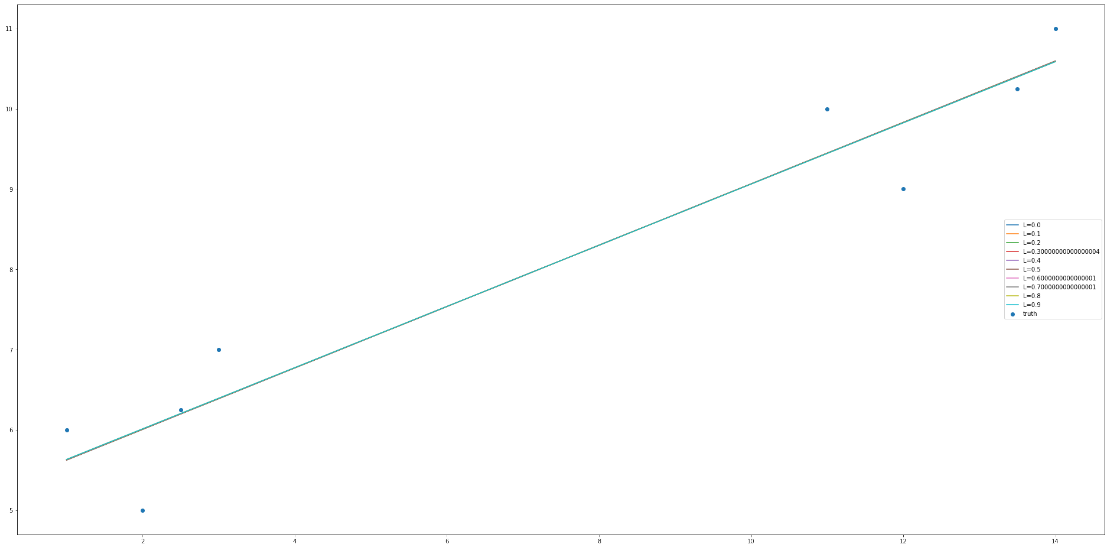


Fig. 3. Task 2.2-Examining different range of $0 \geq \lambda| < 1$ for Ridge Regression

Regrading the assumption which $\lambda > 0$ and as $\theta^2 > 0$, the term $\lambda\theta^2 > 0$. Consequently, Ridge regression needs to shrink the parameters as the positive value of $\lambda\theta^2 > 0$ adds up to $\sum_{t=1}^{n}(y_t - \theta x_t - \theta_0)^2$ to minimize the whole expression. and as result, the bigger λ we have, results in bigger term $\lambda\theta$ which will penalize the parameters even more with a smaller slope θ and bigger intercept θ_0 which results have been examined in a random generated data-set as in Fig. 2 and Fig.3.

Thus our conclusion for models (A),(B),(C) and (D) in 4 is as following:

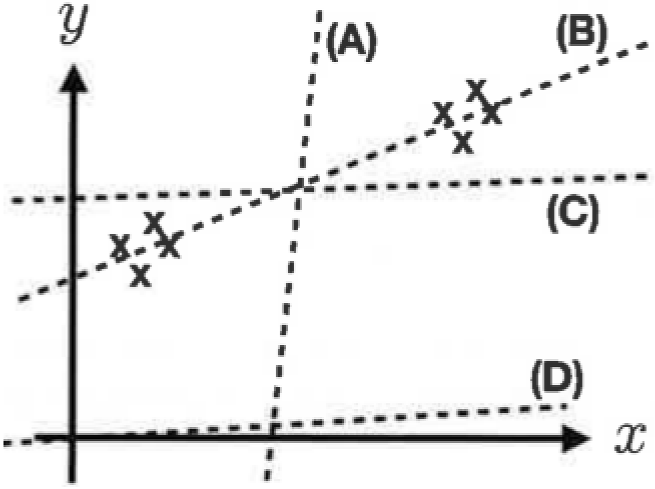


Fig. 4. Task 2.2 models

1. **High** λ : Model (C) can be created by high range of λ values since it has a lower variance, and we can observe parameters θ is approaching to 0 to decrease the model sensitivity to change of the predictor and the intercept θ_0 is penalized to fit the model.
2. **Low** λ : Model (B) can be created by this range of λ values since it has a higher variance fitting the training data tightly, and we can observe parameters θ is not penalized to decrease the model sensitivity and the squared residuals is fairly minimized.
3. **Neither** Models (A) and (D) is in this category. In model (A) we may intuitively observe the parameter θ is even more than slop of most incline line that we can draw with our data; Meanwhile in Ridge regression the parameter is minimized by the sum of squared residuals and the penalty parameter which is even less incline bu the models created by simple linear regression.

Hence it is impossible that model (A) is created by Ridge regression. The model (D) is also not produced by Ride regression since intercept θ_0 is not penalized properly in the range of our data and such an intercept could not be created by Ridge regression.

3 Logistic Regression

3.1 Practical Task 2

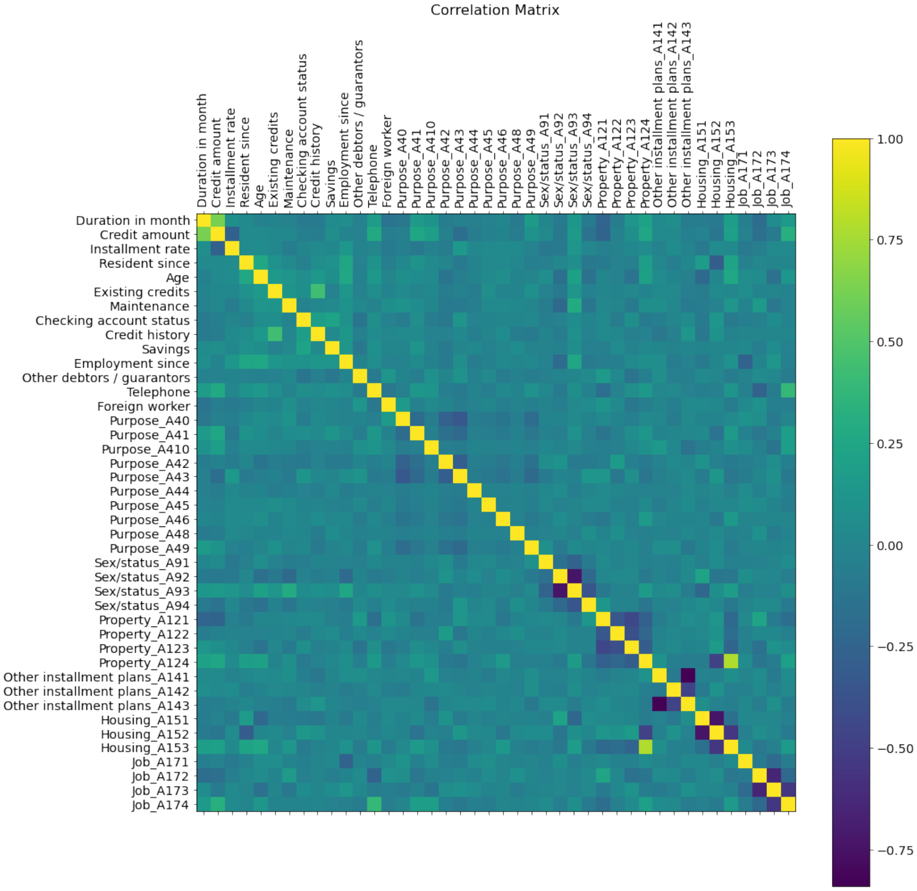


Fig. 5. Task3.1-Features Correlation Matrix

According to the correlation matrix in Fig. 5 and the filter on the highly correlated columns below, the most correlated features are columns *Other installments plans-A141* and *Other installments plans-A143* which corresponds to categories of the **Attribute 14 a.k.a other installments** in the data set. Besides, *Housing-A153* and *Property-A124* **Property** features are also highly co-related, which also makes intuitive sense such that if someones has a free housing, with correlation of 0.779822 we may imply that he ain't got no property. and the rest of correlated features are visible in the filtered results which are provided below. Highly correlated features in training sets can cause the *multi-col-linearity*[2] issue. In such a scenario, the features which are producing the linear model, are not linearly independent, which can cause singularity in solution of the linear regression.

```
----------------------- ABS( CORRELATION ) > 0.5 ---------------------
Other installment plans_A141  Other installment plans_A143   -0.840518
Other installment plans_A143  Other installment plans_A141   -0.840518
Sex/status_A92                Sex/status_A93                 -0.737897
Sex/status_A93                Sex/status_A92                 -0.737897
Housing_A151                  Housing_A152                   -0.735900
Housing_A152                  Housing_A151                   -0.735900
Job_A172                      Job_A173                       -0.652328
Job_A173                      Job_A172                       -0.652328
Housing_A152                  Housing_A153                   -0.548368
Housing_A153                  Housing_A152                   -0.548368
Job_A173                      Job_A174                       -0.543739
Job_A174                      Job_A173                       -0.543739
Duration in month             Credit amount                   0.624714
Credit amount                 Duration in month               0.624714
Property_A124                 Housing_A153                    0.779822
Housing_A153                  Property_A124                   0.779822
```

Regarding Fig.6 and the model metrics, we may conclude model with degree 1 is the best model which does not over-fit our model and has better score.

```
>> LOG:  MAXIMUM RECALL SCORE 0.919047619047619  DEG = 1
>> LOG:  MAXIMUM RECALL SCORE 0.8041666666666667  DEG = 1
>> LOG:  MAXIMUM ACCURACY SCORE 0.7866666666666666  DEG = 1
>> LOG:  MINIMUM MSE SCORE 0.21333333333333335  DEG = 1
```

After tuning the hyper-parameters, we get the following output as the model scores:

```
Finely-Tuned Model Accuracy Score 0.7566666666666667
Finely-Tuned Model Recall Score: 0.8904761904761904
Finely-Tuned Model Percision Score 0.7890295358649789
Finely-Tuned Model MSE 0.24333333333333335
```

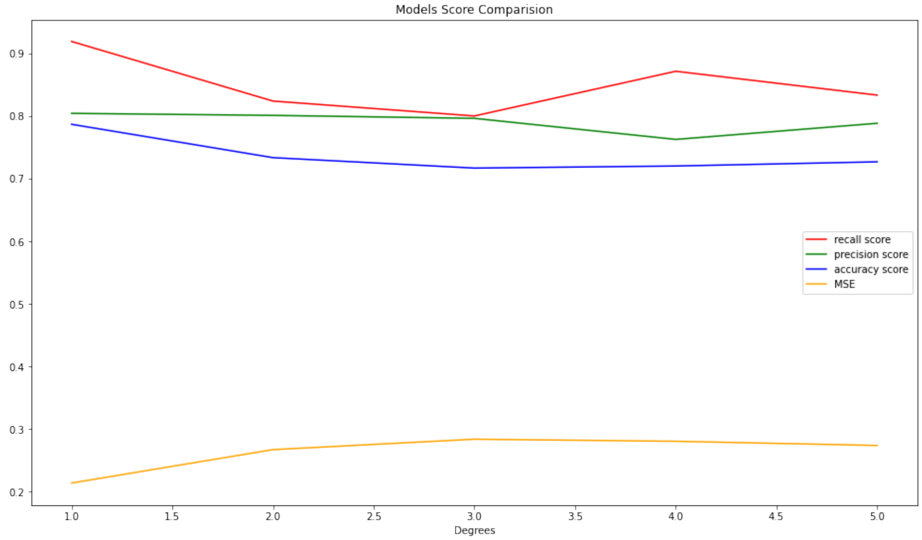


Fig. 6. Task 3.1 - Comparison model metrics in different degrees

Finally, Comparing the tuned model prediction on female and male test results, Model has higher accuracy and less MSE for male applicants , moreover it appears to have more optimistic result on female group due to less precision rate in comparison to male applicants.

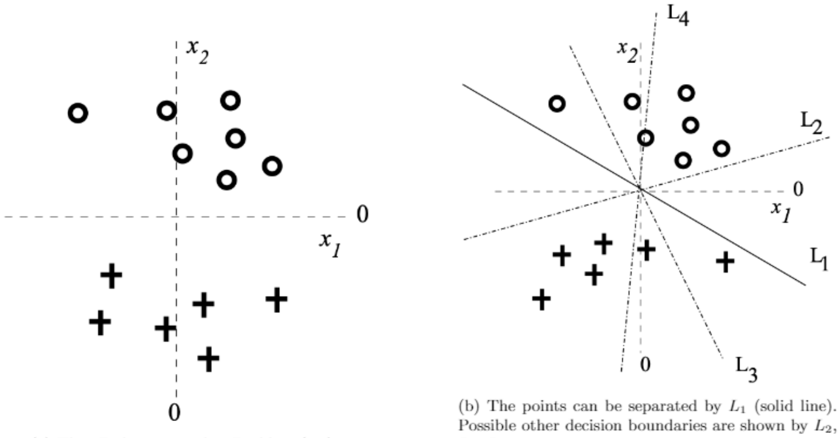
```
>> LOG:  Model accuracy for male applicants test =  0.7853658536585366
>> LOG:  Model MSE for male applicants test =  0.2146341463414634
>> LOG:  Model recall score for male applicants test =  0.8993288590604027
>> LOG:  Model precision score for male applicants test =  0.8220858895705522
---------------------------------------------------------------------------------
>> LOG:  Model accuracy for female applicants test =  0.6947368421052632
>> LOG:  Model MSE for female applicants test =  0.30526315789473685
>> LOG:  Model recall score for female applicants test =  0.8688524590163934
>> LOG:  Model precision score for female applicants test =  0.7162162162162162
```

3.2 Regularization in Logistic Regression

$$\hat{P}(y=1|x;\theta_1,\theta_2)=\frac{1}{1+exp(-\theta_1 x_1-\theta_2 x_2)}$$

$$\hat{\Theta}=\underset{\Theta}{\operatorname{argmax}}(\sum_{i=1}^{n} logP(y_i|x;\theta_0,\theta_1)-\frac{C}{2}\theta_2^2$$

As in the regularization approach only the parameter θ_2 is penalized, as hyper-parameter C approaches to infinity, in order to maximize the aforementioned function, θ_2 would approach to 0. Thus we may conclude the bigger C in



(a) The 2D dataset used in Problem (3.2).

(b) The points can be separated by L_1 (solid line). Possible other decision boundaries are shown by L_2, L_3, L_4.

Fig. 7. Task3.2

regularization results in a smaller θ_2 parameter. Since θ_2 is a coefficient of x_2 in the model which forms the shape of the *decision boundary*, we may consider θ_2 as the slope of the model to x_2 axis.Hence:

1. **Created by Regularization**
   (L_3) Since the x_2 axis slope in less than L_1 model and yet it's greater than 0 appears to be a θ_2 penalized model with aforementioned approach.

2. **Not created by Regularization**
   $(L_2)(L_4)$ Since the x_2 axis slope in the both models is negative, and as we know the penalized x_2 slope can not be negative, it is impossible that these models were created by this regularization.

References

1. Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., Varoquaux, G.: API design for machine learning software: experiences from the scikit-learn project. In: ECML PKDD Workshop: Languages for Data Mining and Machine Learning. pp. 108–122 (2013)
2. Daoud, J.: Multicollinearity and regression analysis. Journal of Physics: Conference Series **949**, 012009 (12 2017). https://doi.org/10.1088/1742-6596/949/1/012009