FLATIRON SCHOOL

Polynomials and Regularisation

Jan 7th 2020

Linear Regression Recap

Task: Take turns with a neighbor, answering the questions below:

- What is Linear Regression?
- How does it work?
- How do we know if the results are good or bad?
- What requirements should your data meet?

5 minutes

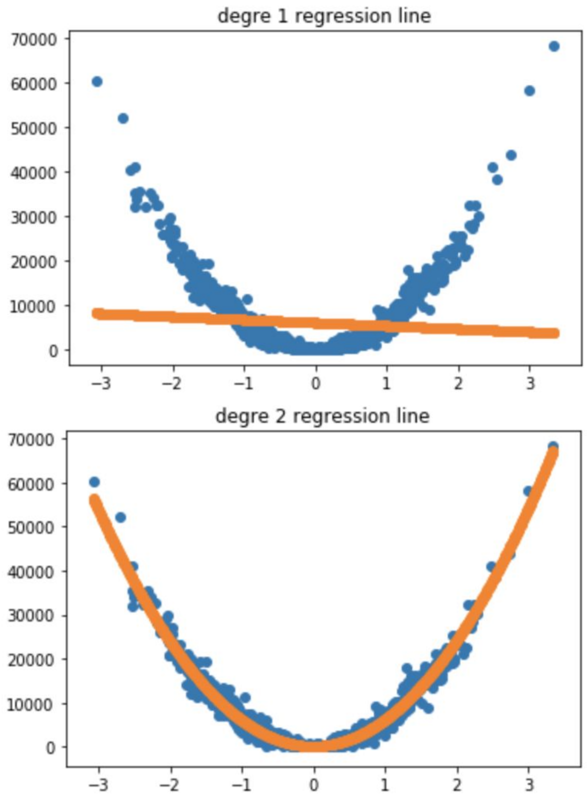
We will then come back to the large group and I'll pick some of you to share your answers with the rest of the class.

Polynomial Regression

- Problems Linear Regression can’t solve on its own
- Polynomial transformation of predictors
- Interactions
- Feature explosion

Polynomial Regression

- Some problems just can't be solved with a straight line
- We need models as complex as our problems in order to generate good predictions
- Example: If $y = x^2$ we need to have x^2 as a feature
- PolynomialFeatures generates



Feature Interactions

- PolynomialFeatures generates the n-way interactions for all your predictions

- Example: If you want the interaction of 3 variables together (AxBxC) you will need a degree 3 polynomial transformation $(A+B+C)^3$

Classifying Polynomials by Degree		
Name	**Degree**	**Example**
Constant	0	-9
Linear	1	$x - 4$
Quadratic	2	$x^2 + 3x - 1$
Cubic	3	$x^3 + 2x^2 + x + 1$
Quartic	4	$2x^4 + x^3 + 3x^2 + 4x - 1$
Quintic	5	$7x^5 + x^4 - x^3 + 3x^2 + 2x - 1$

SQUARE OF SUM

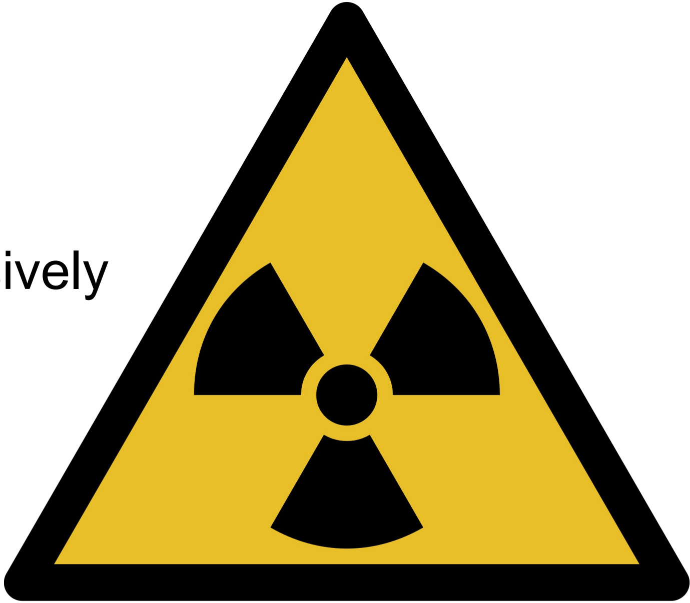
$$(a + b)^2 = a^2 + 2ab + b^2$$

CUBE OF SUM

$$(a + b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$$

Feature Explosion

- This can get out of hand very quickly
- Example: Just 10 Features with a 3rd degree polynomial would lead to 1000 features
- When the number of features is too high:
  - Coefficients become unstable
  - Chance of multicollinearity increases massively
  - Chance of overfitting explodes
  - Dimensionality becomes a course



Polynomial Regression Recap

Task: Take turns with a neighbor, answering the questions below:

- When is polynomial regression useful?
- When could it be dangerous?
- How do you think you could contain the risks?

5 minutes

We will then come back to the large group and I'll pick some of you to share your answers with the rest of the class.

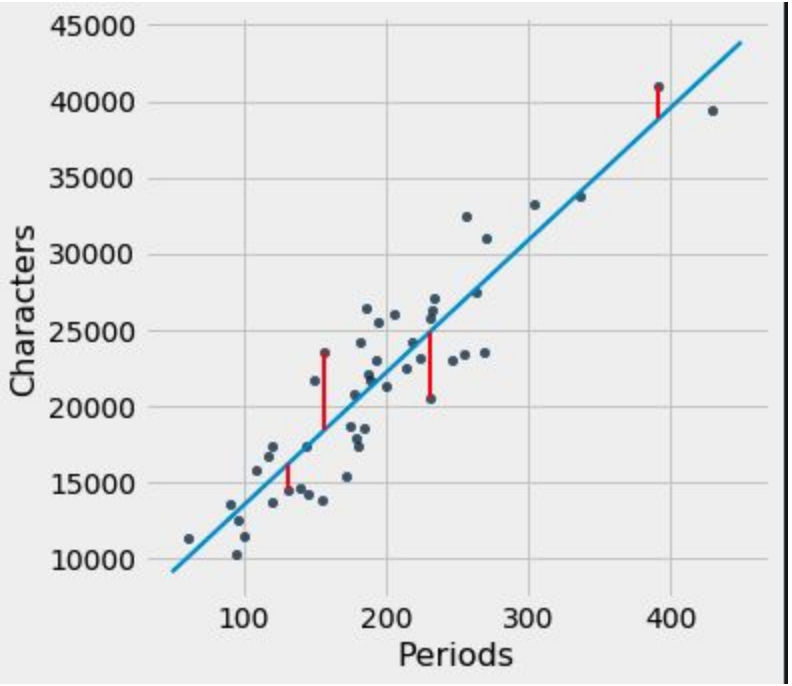
Regularisation

- MSE
- Complexity penalty
- Scaling
- Ridge Regression
- LASSO Regression
- Elastic net
- Bias / Variance tradeoff

Mean Squared Error (MSE)

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

* n is the number of data points
* Y_i represents observed values
* $\hat{Y}_i$ represents predicted values



Complexity Penalty

- Regularisation in Regression means not only trying to minimise MSE but also the size of the coefficients

$$\sum_{i=1}^{M}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{M}\left(y_i - \sum_{j=0}^{p} w_j \times x_{ij}\right)^2 + \lambda \sum_{j=0}^{p} w_j^2$$

Cost function for ridge regression

- Because the size of the coefficients matter **feature scaling** is required.
- Although you will be scaling all datasets, you will use the **patterns from your training** dataset only
-

Ridge Regression - L2 Reg

- Penalty based on the squared coefficients.

- This will lead to reducing largest coefficients first

$$\sum_{i=1}^{M}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{M}\left(y_i - \sum_{j=0}^{p} w_j \times x_{ij}\right)^2 + \lambda \sum_{j=0}^{p} w_j^2$$

Cost function for ridge regression

- It will bring irrelevant features’ coefficients close to zero but not exactly zero

LASSO Regression - L1 Reg

- Penalty based on the absolute value of the coefficients.

- This will lead to reducing most irrelevant coefficients first

$$\sum_{i=1}^{M} (y_i - \hat{y_i})^2 = \sum_{i=1}^{M} \left(y_i - \sum_{j=0}^{p} w_j \times x_{ij} \right)^2 + \lambda \sum_{j=0}^{p} |w_j|$$

Cost function for Lasso regression

- It will bring irrelevant features' coefficients exactly to zero. Hence, we can use LASSO as feature selection tool.
- Because of all of the above we call this type of regression LASSO: Least Absolute Shrinkage and Selection Operator

Elastic net - L2 and L1 Reg

- Combines L2 and L1 penalties in a proportion that you can regulate with a hyperparameter in sklearn

$$(||y - X\beta||^2 + \lambda_2||\beta||^2 + \lambda_1||\beta||_1)$$

- Will bring some coefficients to zero but could also just reduce them if that leads to a better model

Information Criteria

- Measure information loss
- Balances the goodness of fit and the complexity of a model

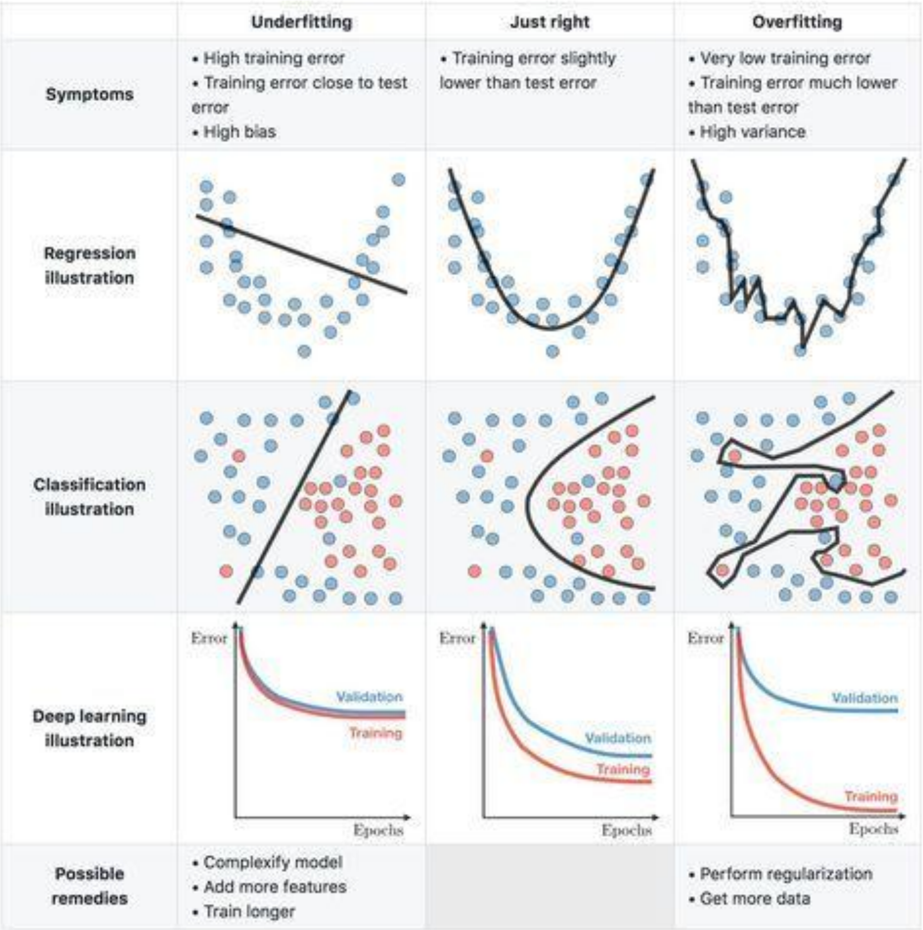
- Akaike Information Criterion $\mathrm{AIC} = 2k - 2\ln(\hat{L})$
- Bayesian Information Criterion $\mathrm{BIC} = \ln(n)k - 2\ln(\widehat{L}).$

- Lower is better

Solving fitting issues

- **Polynomial transformations** will improving performance on the training dataset
- Convenient when previously underfitting

- **Regularisation** will decrease performance in the training dataset
- Convenient when previously overfitting (as it would likely improve performance on the test dataset)



Regularisation Recap

Task: Take turns with a neighbor, answering the questions below:

- When is regularisation useful?
- What assumptions need to be met by your data before you apply it?
- What negative consequences could it have if done wrong?
- How would you choose between different types of regularisation?

5 minutes

We will then come back to the large group and I’ll pick some of you to share your answers with the rest of the class.

Delivering Value

Your job is not to create high performance models

They pay you to **solve problems**

Summary + Exit Ticket

Presented by Dan Sanz