PHYS–467 Machine Learning for Physicists

Regression

October 1, 2021

Exercise 1: Linear Regression in 1D

1. Generate one-dimensional linear data, add Gaussian noise, and plot them.
2. Write a function that takes as input the data matrix X and the labels y, and returns the coefficient and the bias using the closed-form solution.
3. Evaluate the linear regression predictor on a set of test points. Add it to the previous plot.
4. Repeat the last points using the `LinearRegression` function from the Scikit-Learn library.

Exercise 2: Regression in High Dimensions

1. Let X be an $n \times d$ random Gaussian matrix. Compute and plot the spectrum of the matrix $X^\top X$ for dimension $d = 50$ and varying $\alpha = \frac{n}{d}$. How does the rank of this matrix change with α?
2. Add a small regularisation term. How does it affect the spectrum?
3. Plot the behaviour of the test error vs α for high-dimensional (noiseless) linear data. Use $d = 100$ and vary n from 10 to 500. What do you observe for $\alpha < 1$ and $\alpha > 1$?

Exercise 3: Real Data – Predicting Boston House Prices

1. Load the Boston house prices dataset and split it into a training and testing sets.
   Hint: use `train_test_split` from Scikit-Learn.
2. Normalize the features and perform linear regression on the training set. Print the mean squared error and plot the predicted prices vs the actual ones.
3. Vary the training set size n and plot the learning curves (train and test error vs n). Comment.
4. Write a function implementing non-interacting polynomial features. Repeat regression varying the degree from 1 to 6. Plot the results and comment.
5. Add different regularisers to the degree-6 model. Plot the results and find the best value of the regulariser.
   Hint: use `Ridge` from Scikit-Learn.

Exercise 4: Polynomial Regression, Underfitting, and Overfitting

1. Generate quadratic noisy data and try to learn them using linear and polynomial feature regression with a second and a higher-degree polynomial (e.g. 15). Compute and compare the mean squared errors in the three cases. Plot the three predictors, together with the training points. What do you observe?
   Hint: use `PolynomialFeatures` from Scikit-Learn.
2. Repeat the analysis adding a ridge to the high-degree polynomial case. Vary the strength of the regulariser.What happens?