Exercise 2

Prof. Dr. Amr Alanwar

October 2024

Pandas (5 Points)

- **Dataset Exploration:** Download Gasprices.csv. This dataset contains information about the sales of gas stations across a city along with other attributes. You will analyze this dataset using pandas library and plot some interesting information using the matplotlib library.
  - Load the data using pandas.
  - Summarize each **NUMERIC** field in the data, i.e., mean, average, etc.
  - Group data by the field `Name`.
    - Find the average price, average income, and average number of pumps for each group.
    - Use a boxplot that visualizes the statistical information about (price, pumps, gasoline).
    - Use the Price and Income features to plot a prediction line similar to the first exercise. Normalize the Income (implement this yourself) and plot the line again. Comment on the difference between the two plots.

Linear Regression via Normal Equations (5 Points)

In this exercise, you will implement (multiple) linear regression using Normal Equations. The learning algorithm is below.

- Reuse the Gasprices.csv dataset. Load it as `Xdata`.
- Choose columns that help with prediction (i.e., contain useful information). Drop irrelevant columns, and explain your reasoning for choosing or dropping any column.
- Split your dataset `Xdata`, `Ydata` into `Xtrain`, `Ytrain`, and `Xtest`, `Ytest` (randomly assign 80% to `Xtrain`, `Ytrain` and the remaining 20% to `Xtest`, `ytest`).
- Implement the `learn-linreg-NormEq` algorithm and learn a parameter vector β using the `Xtrain` set. You need to learn a model to predict the sales price of houses, i.e., `ytest`.
- Line 6 of the `learn-linreg-NormEq` algorithm uses `SOLVE-SLE`. You must replace `SOLVE-SLE` with the following options (implement this yourself):
  - Gaussian elimination
  - Cholesky decomposition
  - QR decomposition
- Perform predictions $\hat{y}$ on the test dataset `Xtest` using the parameters learned in steps 5 and 6. *[Hint: You will have three different prediction models based on the replacement function from step 6.]*
- The final step is to find how close these three models are to the original values.
  - Plot the residual $\varepsilon = |ytest - \hat{y}|$ versus the true value of `ytest` for each model.

– Find the average residual $\varepsilon = |ytest - \hat{y}|$ for each model.

– Compute the root-mean-square error (RMSE) as follows:

$$\text{RMSE} = \sqrt{\frac{\sum_{n=1}^{N}(ytest(n) - \hat{y}(n))^2}{N}}$$

for each model.

0.1 ANNEX

- You can use `numpy` or `scipy` built-in methods for linear algebra operations.
- You can use `pandas` for reading and processing data.
- You can use `matplotlib` for plotting.
- You should not use any machine learning library (e.g., `scikit-learn`) for solving the problem. If you use them, you will not receive any points for the task.

Algorithm 1 Learn-LinReg-NormEq

procedure LEARN-LINREG-NORMEQ($D_{\text{train}} := \{(x_1, y_1), \ldots, (x_N, y_N)\}$)
  $X := (x_1, x_2, \ldots, x_N)^T$
  $y := (y_1, y_2, \ldots, y_N)^T$
  $A := X^T X$
  $b := X^T y$
  $\hat{\beta} :=$ SOLVE-SLE(A, b)
  return $\hat{\beta}$
end procedure