3 Linear Regression

3.1 Practical Session

Let us consider the `iris_dataset`. In the dataset we have data regarding specific species of flowers (Figure 3.1):

- Sepal length;
- Sepal width;
- Petal length;
- Petal width;
- Species (Iris setosa, Iris virginica e Iris versicolor).

in the specific, we have $N = 150$ total samples (50 per type).

At first, we want to predict the **petal width** of a specific kind of *Iris setosa* by using the **petal length**. This can be considered a regression problem where we consider as feature x_n the petal length and as target t_n the petal width. In order to provide a prediction $\hat{t}_n$ for the target t_n, we will consider:

- Hypothesis space: $\hat{t}_n = f(x_n, w) = w_0 + x_n w_1$;
- Loss measure: $J(w, x_n, t_n) = RSS(w) = \sum_n (\hat{t}_n - t_n)^2$;
- Optimization method: Least Square (LS) method.

where $w \in \mathbb{R}^M, M = 2$.

3.1.1 Data Pre-processing

We load the data into the `MATLAB` workspace:

```
1 load iris_dataset.mat;
```

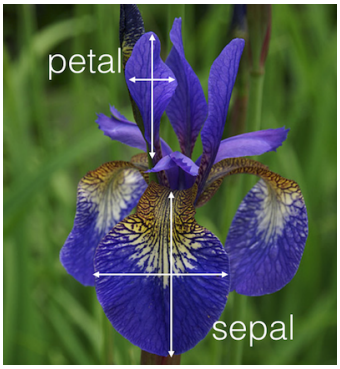


Figure 3.1: Image of an Iris flower.

Before even starting the process of analyzing data, one should plot the considered data to inspect them (if possible):

```
figure();
gplotmatrix(irisInputs');
x = irisInputs(3,:)';
t = irisInputs(4,:)';
```

Once we inspected the data, we should operate some pre-processing procedures. On a generic dataset one should perform:

- shuffling;
- remove inconsistent data;
- remove outliers;
- normalize or standardize data;
- fill missing data.

For instance in the Iris dataset there has been some problem in the transcription of the original dataset and some works has been tested on a different dataset.[1]. Thus one might want to remove those data which have been erroneously reported or correct them if the original values are available.

In this case, we simply normalize the data (input and target) by using the function `zscore()` which operates the following operation:

$$s \leftarrow \frac{s - \bar{s}}{S} \text{ or } s \leftarrow \frac{s - \bar{s}}{\max_n\{s_n\} - \min_n\{s_n\}},$$

[1] `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=771092`

where $\bar{s} := \frac{\sum_n s_n}{N}$ is the sample mean for a dataset s and $S := \sqrt{\frac{\sum_n (s_n - \bar{s})^2}{N-1}}$ is the estimates of the standard deviation. In MATLAB , the line corresponding to the normalization with the standard deviation are:

```
x = zscore(x);
t = zscore(t);
figure();
plot(x,t,'bo');
```

3.1.2 Linear Regression Options

Once we pre-processed the data, we might resort to use the MATLAB fitting toolbox to find solve the regression problem. This toolbox supports at most 2 input variables and is able to predict a single scalar output. It is possible to consider different hypothesis spaces. For instance one might consider a polynomial regression or introduce a specific basis functions $\phi_i(\mathbf{x})$ over the available input $\mathbf{x}$.

In the recap of a fitting process, it is possible to inspect some of the figure of merits that allow us to understand if the fitting we considered was valuable:

- Residual Sum of Squares: (`sse` in MATLAB) $RSS(w) = \sum_n (\hat{t}_n - t_n)^2$, telling us how much of the prediction differs from the true value;
- Coefficient of determination: (`rsquare`) $R^2 = 1 - \frac{RSS(w)}{\sum_n (\bar{t} - t_n)^2}$ telling us how the fraction of the variance of the data explained by the model (how much better we are doing w.r.t. just using the mean of the target $\bar{t}$). In space with a single feature this is equal to the correlation coeffiecient between the input and the output;
- Degrees of Freedom: (`dfe`) $dfe = N - M$ telling us how much our model is flexible in fitting the data;
- Adjusted coefficient of determination (`adjrsquare`) $R^2_{adj} = R^2 \frac{N}{dfe}$ the coefficient of determination corrected w.r.t. how much flexibility the model has;
- Root Mean Square Error: (`rmse`) $RMSE = \sqrt{\frac{RSS(w)}{N}}$ telling approximately how much error we get on a predicted data over the training set (i.e., a normalized version of the RSS).

where the mean of the targets is $\bar{t} = \frac{\sum_n t_n}{N}$.

Moreover, it is possible to have a confidence interval for the estimated coefficients w_0 and w_1 (called a and b in the toolbox). In fact, it is possible to show that, under the assumption that the observations t_n are i.i.d. and satisfies $t_n = w_0 + \sum_j w_j x_{nj} + \varepsilon$,

where ε is a Gaussian noise with zero mean and variance σ^2 (i.e., the data are generated by a linear model with noise), the computed coefficients $\hat{w}_j$ are distributed as follows:

$$\frac{\hat{w}_j - w_j}{\hat{\sigma}\sqrt{v_j}} \sim t_{N-M-1}$$

where w_j is the true parameter, $\hat{\sigma}$ is the unbiased estimated for the target variance, i.e., $\hat{\sigma}^2 = \frac{\sum_n (t_n - \bar{t}_n)^2}{N-M-1}$, v_j is the j-th diagonal element of the matrix $(X^T X)^{-1}$ and t_{N-M-1} is the t-student distribution with $N - M - 1$ degrees of freedom.

Another possibility is to call the function `fit()`. In this case, we need to specify the input and output variables and the name of the model parameters. In the MATLAB language we have:

```
fit_specifications = fittype( {'1', 'x'}, 'independent', 'x', '
   dependent', 't', 'coefficients', {'w0', 'w1'} );
[fitresult, gof] = fit( x, t, fit_specifications);
```

where we specified with `fittype()` the model we want to fit.[2]

The results of the fitting process is:

```
  Linear model:
     fitresult(x) = w0 + w1*x
     Coefficients (with 95% confidence bounds):
       w0 =   1.107e-15  (-0.04377, 0.04377)
       w1 =      0.9628  (0.9188, 1.007)

           sse: 10.8917
       rsquare: 0.9269
           dfe: 148
    adjrsquare: 0.9264
          rmse: 0.2713
```

Also here we have information about the confidence intervals of the parameters and about common goodness of fit values, to evaluate if the chosen model is consistent with the real relationship existing between input and target.

Similarly, we can use the function `fitlm()`, which is more general to the previously seen ones, since it allow us to perform even multiple regression (i.e., with more than two inputs). An analogous code for regression is:

```
ls_model = fitlm(x,t);
```

[2] For a wider choice of fitting models for `fittype()` see `http://it.mathworks.com/help/curvefit/list-of-library-models-for-curve-and-surface-fitting.html#btbcvnl`.

In this case the output is:

```
Linear regression model:
    y ~ 1 + x1

Estimated Coefficients:
              Estimate      SE         tStat      pValue
              __________  ________  __________  __________

(Intercept)   1.1073e-15  0.02215   4.9989e-14           1
x1            0.96276     0.022224  43.32       5.7767e-86


Number of observations: 150, Error degrees of freedom: 148
Root Mean Squared Error: 0.271
R-squared: 0.927,  Adjusted R-Squared 0.926
F-statistic vs. constant model: 1.88e+03, p-value = 5.78e-86
```

where here we have also the information about the F-statistic, which consider the following hypothesis test:

$$H_0 : w_0 = w_1 = \ldots = w_M = 0 \qquad \text{vs.} \qquad H_1 : \exists w_j \neq 0.$$

The F-statistic can be computed and is distributed as follows:

$$F = \frac{dfe}{M-1} \frac{\sum_n (\hat{t}_n - t_n)^2 - RSS(w)}{RSS(w)} \sim F_{M-1,N-M-1} \tag{3.1}$$

where $F_{M-1,N-M-1}$ is the Fisher-Snedecor distribution with parameters $M-1$ and $N-M-1$.

At last, we can implement by scratch the function to perform LS fitting (for instance by using the function `pinv()`):

```
n_sample = length(x);
Phi = [ones(n_sample,1) x];
mpinv = pinv(Phi' * Phi) * Phi';
w = mpinv * t;
```

By using `pinv()` (as well as by using the previous regression tools) we are implicitly using the default tolerance for the eigenvalues of a matrix `A` which has value: `max(size(A)) * norm(A) * eps`.

Remark 1. *Here we considered the feature matrix `Phi` with a constant term and a linear term. This matrix can be expanded by adding new columns, e.g., if we want to consider also a quadratic term we write:*

```
Phi = [ones(n_sample,1) x x.^2];
```

Clearly, by using this last option we are more aware of the tools we use (and we are faster in the computation of the solution), but we have to compute by hand the figures of merit to understand if the regression is valuable, e.g., to compute R^2 we have to execute the following:

```
hat_t = Phi * w;
bar_t = mean(t);
RSS = sum((t-hat_t).^2);
R_squared = 1 - RSS / sum((t-bar_t).^2);
```

3.1.3 Regularization

If we need to mitigate over-fitting effects in a model we might resort to some regularization techniques, like Ridge regression or Lasso regression. In MATLAB ridge regression with a specific regularization parameter λ or lasso regression are obtained by:

```
lambda = 10^(-10);
ridge_coeff = ridge(t, Phi, lambda);

[lasso_coeff, lasso_fit] = lasso(Phi, t);
```

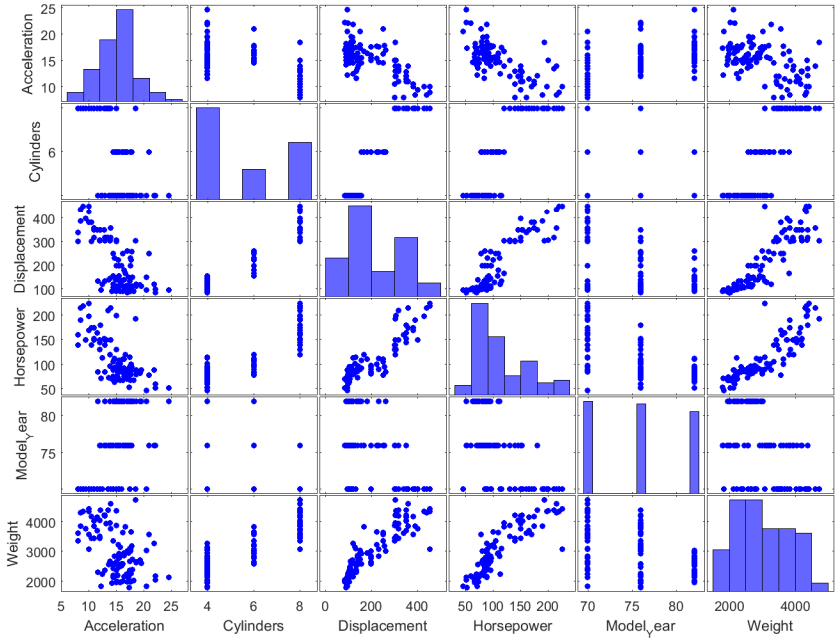
where the ridge regression returns only the coefficients, in this case w_0 and w_1, and the lasso one provides you solutions for different values of the regularization parameter λ.

3.2 Exercises

Exercise 3.1

Consider the Car dataset (`load carsmall`). Write a MATLAB script able to preprocess the data for regression. In the specific, the task one want to perform is to predict the Acceleration of a new car, given all the other features.

Visually select the most appropriate variables one might use to perform the task by looking at the scatterplot in the following figure and prepare them for a linear regression. Assume that the data are contained in a matrix `X` where each row is a car and each column a feature.



Try to guess the sign of the coefficients for each variable you included in the model.

Exercise 3.2

Consider the Car dataset (`load carsmall`) and a linear regression model $Acceleration \sim Cylinder + Displacement + Horsepower$ (all column vectors) which provides the following results:

```
Linear regression model:
    y ~ 1 + x1 + x2 + x3
Estimated Coefficients:
                   Estimate       SE        tStat       pValue
                   ________    _________    _________    _________
    (Intercept)    -0.01539    0.070813    -0.21733      0.82842
    x1              0.17692     0.23198     0.76267      0.44755
    x2             -0.53443     0.28783     -1.8568     0.066443
    x3             -0.35339     0.17506     -2.0186     0.046343
Number of observations: 99, Error degrees of freedom: 95
Root Mean Squared Error: 0.704
R-squared: 0.51,  Adjusted R-Squared 0.495
F-statistic vs. constant model: 33, p-value = 1.04e-14
```

FITLM: fit linear model

51% data was explained by the model

Write the command that most likely generated these results and answer to the following questions:

1. Do you think that all the features are significant?
2. Do you think that at least one of the features is significant?
3. How much is the RSS for this model?
4. How much variance is explained by the model?
5. How much error is this model making on average on a new data point?

Exercise 3.3

Consider the Iris dataset. Write a `MATLAB` script which is able to predict the petal width by using all the other features as input.

Comment on the parameters we would like to introduce or exclude from the prediction process. Does this model is better than the one trained with a single input?

Exercise 3.4

Consider the Iris dataset. Write a `MATLAB` script which is able to predict the petal width by using all the other features as input (the data are in the variable `irisInput` and are structured with samples on the columns and features on the rows, the variables are in order Sepal length, Sepal width, Petal length and Petal width).

Consider the following `MATLAB` commands:

```
[lasso_coeff, lasso_fit] = lasso(x, t);
ridge_coeff = ridge(t, x, 0.0001);
```

What is the difference w.r.t. the regular linear regression? What are the output of these commands?

Consider a set of parameters λ for the lasso regression increasing with the index. How do you expect the plot `plot(lasso_fit.MSE)` to be?

Exercise 3.5

What might be the problems of this linear model

```
Linear regression model:
    y ~ 1 + x1 + x2
Estimated Coefficients:
             Estimate       SE          tStat        pValue
```

```
5
6  (Intercept)   0.0510818   0.002510709        20.345    3.068974e-77
7  x1            2.9975415   0.003157039        949.47               0
8  x2            2689104.7   327035634.0        0.0082     0.993440980
9  Number of observations: 1000, Error degrees of freedom: 997
10 Root Mean Squared Error: 0.0295
11 R-squared: 0.999,  Adjusted R-Squared 0.999
12 F-statistic vs. constant model: 4.51e+05, p-value = 0
```

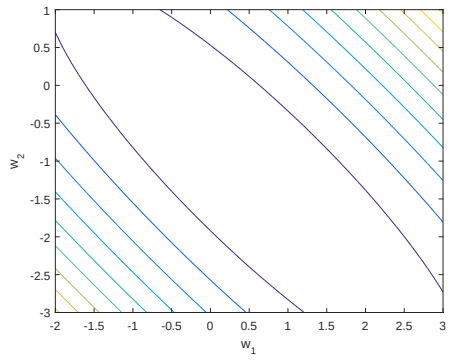
Do you think that all the considered features are significant for the problem?

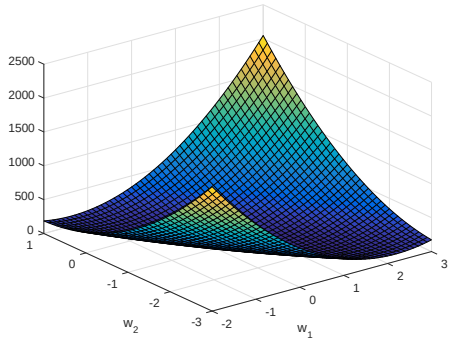
Exercise 3.6

Consider the `carsmall` dataset. Assume the data are structured in a matrix `X` containing the relevant features (weight and displacement, rows are samples and column are features) to predict the acceleration, stored in a vector `t`.

Plot the loss (RSS) as a function of two parameters $w_1 \in [-2, 3]$ and $w_2 \in [-3, 1]$. Use the function `contour` and/or the `surf` one.

Assume that the plot resulted in the following two figures:





what is the parameter one would get if she/he performed linear regression on this dataset?

3.3 Questions

Exercise 3.7

Given the expression:

$$S = f(TV, R, N),$$

where S is the amount of sales revenue, TV, R and N are the amount of money spent on advertisements on TV programs, radio and newspapers, respectively, explain what are the:

1. Response; *S*
2. Independent variables; *TV,R,N*
3. Features; *All the input I have*
4. Model, *f()*

Exercise 3.8

Explain why the following problems can or cannot be addressed by Machine Learning (ML) techniques:

1. Partition a set of employees of a large company;
2. Fortune-telling a person information about her/his personal life;
3. Determine the truthfulness of a first order logic formula;
4. Compute the stress on a structure given its physical model;
5. Provide temperature predictions.

supervised:
unsupervised:
RL:

In the case the problem can be addressed by ML, provide a suggestion for the technique you would use to solve the problem.

[SEMI-supervised]
[Active Learning]

Exercise 3.9

Categorize the following ML problems:

1. Predicting housing prices for real estate; *regression since the price are ordered*
2. Identify inside trading among stock market exchange; *classification: if someone is a potential trader or not*
3. Detect interesting features from an image; *we dont have anything to predict we just want to get a specific pattern from what we have*
4. Determine which bird species is/are on a given audio recording; *classication*
5. Teach a robot to play air hockey; *RL*
6. Predicting tastes in shopping/streaming; *supervised learning*

7. Recognise handwritten digits;
8. Pricing goods for an e-commerce website.

For each one of them suggest a set of features which might be useful to solve the problem and a method to solve it.

Exercise 3.10

Why is linear regression important to understand? Select all that apply and justify your choice:

1. The linear model is often correct;
2. Linear regression is extensible and can be used to capture nonlinear effects;
3. Simple methods can outperform more complex ones if the data are noisy;
4. Understanding simpler methods sheds light on more complex ones;
5. A fast way of solving them is available.

Exercise 3.11

Consider a generic regression model. Tell if one should consider the LS method as a viable option in the following 4 different situations. Motivate your answer.

1. Small number of parameters;
2. The loss function is $L(\mathbf{w}|x_n, t_n) = |y(x_n, \mathbf{w}) - t_n|$;
3. Huge number of samples;
4. The loss function is $L(\mathbf{w}|x_n, t_n) = \begin{cases} (y(x_n, \mathbf{w}) - t_n)^2 & \text{if} |y(x_n, \mathbf{w}) - t_n| < \delta \\ |y(x_n, \mathbf{w}) - t_n| & \text{if } |y(x_n, \mathbf{w}) - t_n| > \delta \end{cases}$

Exercise 3.12

Which of the following statements are true? Select all that apply and provide a motivation for your choice:

1. A 95% confidence interval is a random interval that contains the true parameter 95% of the time;
2. The true parameter is a random value that has 95% chance of falling in the 95% confidence interval

3. I perform a linear regression and get a 95% confidence interval from 0.4 to 0.5. There is a 95% probability that the true parameter is between 0.4 and 0.5.

4. The true parameter (unknown to me) is 0.5. If I sample data and construct a 95% confidence interval, the interval will contain 0.5 95% of the time.

Exercise 3.13

Consider a linear regression with input x, target t and optimal parameter θ^*.

1. What happens if we consider as input variables x and $2x$?

2. What we expect on the uncertainty about the parameters we get by considering as input variables x and $2x$?

3. Provide a technique to solve the problem.

4. What happens if we consider as input variables x and x^2?

* Exercise 3.14

Consider a data set in which each data point (x_n, t_n) is associated with a weighting factor $r_n > 0$, so that the error function becomes:

$$J(\mathbf{w}) = \frac{1}{2N} \sum_{n=1}^{N} r_n (\mathbf{w}^T x_n - t_n)^2$$

Find an expression for the solution that minimizes this error function. Give two alternative interpretations of the weighted sum-of-squares error function in terms of (i) data dependent noise variance and (ii) replicated data points.

Exercise 3.15

Consider an initial parameter $\mathbf{w}^{(0)} = [0\ 0\ 1]$ and a loss function of the form:

$$J(\mathbf{w}) = \frac{1}{2N} \sum_{n=1}^{N} (\mathbf{w}^T x_n - t_n)^2.$$

Derive the update given from the gradient descent for the datum $x_1 = [1\ 3\ 2]$, $t_1 = 4$ and a learning rate $\alpha = 0.3$.

What changes if we want to perform a batch update with $K = 10$ data?

Exercise 3.16

Which of the following indicates a fairly strong relationship between an input variable X and the output variable Y of a linear regression? Why?

1. $R^2 = 0.9$ in a single dimensional linear regression;
2. $R^2 = 0.9$ in a multiple linear regression;
3. The p-value for the null hypothesis $H_0 : \ w_1 = 0$ is 0.0001 in a multiple linear regression;
4. The t-statistic for the null hypothesis $H_0 : \ w_1 = 0$ is 30.

Exercise 3.17

After performing Ridge regression on a dataset with $\lambda = 10^{-5}$ we get one of the following one set of eigenvalues for the matrix $(\Phi^T\Phi + \lambda I)$:

1. $\Lambda = \{0.00000000178, \ 0.014, \ 12\}$;
2. $\Lambda = \{0.0000178, \ -0.014, \ 991\}$;
3. $\Lambda = \{0.0000178, \ 0.014, \ 991\}$;
4. $\Lambda = \{0.0000178, \ 0.0000178, \ 991\}$;

Explain whether these sets are plausible solutions or not.

Exercise 3.18

We run a linear regression and the slope estimate is $\hat{w}_k = 0.5$ with estimated standard error of $\hat{\sigma} v_k = 0.2$. What is the largest value of w for which we would NOT reject the null hypothesis that $\hat{w}_1 = w$? (assume normal approximation to t distribution, and that we are using the $\alpha = 5\%$ significance level for a two-sided test.

Exercise 3.19

Which of the following statements are true? Provide motivations of your answers.

1. The estimate w_1 in a linear regression for many variables (i.e., a regression with many predictors in addition to x_1) is usually a more reliable measure of a causal relationship than w_1 from a univariate regression on X_1;
2. One advantage of using linear models is that the true regression function is often linear;
3. If the F-statistic is significant, all of the predictors have statistically significant effects;

4. In a linear regression with several variables, a variable has a positive regression coefficient if and only if its correlation with the response is positive.

Exercise 3.20

Let us assume that the solution with LS method of a regression problem has result

$$\hat{t} = 5 + 4x$$

We would like to repeat the same regression with a Gaussian Bayesian prior over the parameter space $[w_0 w_1]$ with mean $\mu = [3, 2]^T$ and covariance matrix $\sigma^2 = I_2$ (identity matrix of order 2).

Which of the following are consistent solutions to the previous regression problem with the previously specified Bayesian prior?

1. $w = [5, 4]$;
2. $w = [4, 3]$;
3. $w = [6, 5]$;
4. $w = [2, 3]$.

* Exercise 3.21

Derive the analytical solution for the *Ridge Regression*. We remember that it considers as loss function the following one:

$$J(\mathbf{w}) = \sum_{n=1}^{N} (\mathbf{w}^T x_n - t_n)^2 + \lambda ||\mathbf{w}||_2^2$$

Derive the gradient descent scheme for the Ridge Regression.