INF 5860 Weekly exercises on linear regression

These exercises can give you a hint about how exercises for the written exam can be.

Exercise 1:

a) What is the loss function for linear regression? Describe by words and formula.

b) Why would we use an iterative algorithm for the linear regression problem?

c) What can happen if the learning rate is too high or too low?

d) How does the gradient descent algorithm update the θ's?

Exercise 2:

You are given a vector a measurements x and true values y

$$x = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \; y = \begin{bmatrix} 1.5 \\ 2 \\ 2.5 \end{bmatrix}$$

a) Plot y and x as points.

b) If we start with θ^0=0 and θ^1=0, what is the initial value for the loss function?

c) Compute the next estimate of θ^0 and θ^1, after 1 iteration of gradient descent.

Programming exercise – Implementing linear regression

In this exercise, you will implement linear regression. You should implement the following functions. nosetests will be available later, but you can start the implementation.

```
def linregpredict(X, thetavec):
  # Use the training data X and the current estimate thetavec to predict yhat
  # Hint: Use .dot to predict all samples in one operation
  # Fill in…..
  return yhat

def linregloss(y,yhat):
  # Compute the loss as the mean square error summed over all samples in y and yhat.
  # Hint: use np.sum() to sum over all m samples
  # Fill in….
   return mseloss

def gradientdescent(yhat, y, epsilon, thetavec, X):
  # Use gradient descent with learning rate epsilon the predict the next value of thetavec
  # You can use a loop over the indexes in thetavec
  # Fill in……
  return newthetavec

def normalequations(X, y, thetavec):
  # Use the normal equation to provide an alternative estimate of thetavec
  return thetavec
```

X is a (mxn+1)-matrix where each row has n+1 features values (x^0=0), and there are m rows.
y is a mx1-matrix.
thetavec is a nx1 matrix.

A main-program and test data will be provided.
Use the simple test data to develop and test the program:
n=1
Initialize: theta0 = 0, theta1 = 0

$$Xtest = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} \quad y = \begin{bmatrix} 1 \\ 2 \\ 2.5 \end{bmatrix}$$

Compare the gradient descent estimate to the estimate using the normal equations.