1 Curve fitting / Linear regression

Remember: Linear does **not** mean linear in x but linear in w!
Given $D = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)\}$, $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$ choose $f : \mathbb{R}^D \to \mathbb{R}$ to predict $\hat{y}$ from $\mathbf{x}$

General: $$f(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}) = \sum_{d=1}^{D} w_d \phi_d(\mathbf{x})$$

Identity (vector): $\boldsymbol{\varphi}(\mathbf{x}) = \mathbf{x}$ $\rightarrow$ $$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} = \sum_{d=1}^{D} w_d x_d$$

Polynomial (scalar): $\boldsymbol{\varphi}(x) = (1, x, x^2, x^3, \dots)$ $\rightarrow$ $$f(x) = \mathbf{w}^T \mathbf{x} = \sum_{j=0}^{M} w_j x^j$$

You could also use Fourier basis, Wavelets etc.

Task 1.1 Least Squares

Given are $N = 10$ observations of the process

$$y(x) = \sin(2\pi x) + e \qquad \text{with} \qquad e \sim \mathcal{N}\left(0, \sigma_e^2 = 0.18\right)$$

as training data, where x is evenly spaced in the interval $[0, 1]$. Estimate the coefficients $\mathbf{w} = (w_0, \dots, w_M)^{\mathrm{T}}$ of the polynomial $\hat{y}(x, \mathbf{w})$. Our goal is to use the polynomial to predict an observation of y given a new input x as follows:

$$\hat{y}(x, \mathbf{w}) = \sum_{j=0}^{M} w_j x^j = \mathbf{w}^{\mathrm{T}} \boldsymbol{\varphi}(x)$$
$$\boldsymbol{\varphi}(x) = \begin{bmatrix} x^0 & x^1 & \dots & x^M \end{bmatrix}^{\mathrm{T}}$$

Determine a system of equations for the coefficients $\mathbf{w}$ by minimizing the error function

$$\begin{aligned} E(\mathbf{w}) &= \frac{1}{2} \sum_{n=1}^{N} \left(\hat{y}(x_n, \mathbf{w}) - y_n\right)^2 \\ &= \frac{1}{2} \left\| \boldsymbol{\Phi}(\mathbf{x}) \mathbf{w} - \mathbf{y} \right\|^2 . \end{aligned}$$

- How do you find a possible minimum?
- How is $\mathbf{x}$, $\boldsymbol{\Phi}(\mathbf{x})$ and $\mathbf{y}$ defined?

Hints:

- The result can be expressed using matrices: $\mathbf{A}\mathbf{w} = \mathbf{v}$
- Can you express the solution such that it depends on the data matrix $\boldsymbol{\Phi} = \boldsymbol{\Phi}(\mathbf{x})$?

Task 1.2 Least Squares (Code)

We now want to visualize the regression for different polynomial orders. The code to plot the result is already written for you. However, it calls a function `get_weight_vector` which must be written by you.

a) Figure out, how you can determine the data matrix $\mathbf{X}$.

b) Write the code for the function `get_weight_vector`. What do the arguments mean of this function? Which shape do they have?
c) Is it possible to estimate a curve that hits every training point? When should this happen? Does this happen? If not, do you have an idea why? Test this also with $N = 15$.

Hints:

- If you are not familiar with python, this, this and/or this tutorial might help you to get started
- Look at the documentation of the `polyvander` function (doc)
  - Try to relate it to the ϕ in the introduction
- To invert the matrix, you can use `np.linalg.inv` (doc)
- To get the matrix-matrix product, you need to use the `matmul` function (doc)
- To get the transposed matrix, you can simply access its `T` property

Task 1.3 RMSE (Code)

Generate 100 new test samples for x in the range 0 to 1 and calculate the RMSE between those samples. Calculate the RMSE for the training and test data. Let the linear regression order M start from 0. Choose the highest regression order such, that the RMSE goes to zero. Does the RMSE goes to zero for the training or test data?
Note: The coefficients should still be estimated using the old number of samples.

Task 1.4 Regularization

Determine a system of equations for the coefficients $\mathbf{w}$ by minimizing the error function

$$\widetilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(\hat{y}\left(x_n, \mathbf{w}\right) - y_n\right)^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2.$$

Hint: Compared to the previous task, we just have a slightly different cost function. The steps are basically the same.

Task 1.5 Regularization (Code)

Add the regularization to the function you have written before and add an additional slider to control it. What are reasonable values to try for λ (Linear range?)?
Additional notes:

- `lambda` is a reserved keyword in python (for an anonymous function) so please use another name for the variable
- Can you now increase the polynomial order?
- How does the regularization influence stability of the matrix inversion?

Task 1.6 RMSE (Code)

Use the RMSE function to plot the RMSE for different M as a function of $\ln \lambda$. Can you reproduce the plot from the lecture?

Task 1.7

Show that maximum likelihood (ML) estimation of $\mathbf{w}$ is equal to the least squares solution. Assume that the training data points are i.i.d. and the probability for one point is $p\left(y_n|x_n, \mathbf{w}\right) = \mathcal{N}\left(y_n; \hat{y}\left(x_n, \mathbf{w}\right), \sigma_e^2\right)$

Task 1.8

Show that maximum a posteriori (MAP) estimation of $\mathbf{w}$ is equal to the regularized least squares solution when using the following prior distribution for $\mathbf{w}$:

$$p(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}; \mathbf{0}, \sigma_w^2 \mathbf{I}\right). \tag{1}$$

Hint: The posterior distribution $p(\mathbf{w}|\mathbf{x}, \mathbf{y})$ can be computed using Bayes' theorem

$$p(\mathbf{w}|\mathbf{x}, \mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{w}, \mathbf{x})p(\mathbf{w}|\mathbf{x})}{p(\mathbf{y}|\mathbf{x})} = \frac{p(\mathbf{y}|\mathbf{w}, \mathbf{x})p(\mathbf{w})}{p(\mathbf{y}|\mathbf{x})}. \tag{2}$$

As the denominator is independent of $\mathbf{w}$ we only have to maximize

$$p(\mathbf{w}|\mathbf{x}, \mathbf{y}) \propto p(\mathbf{y}|\mathbf{w}, \mathbf{x})p(\mathbf{w}). \tag{3}$$

Addition

This solution is identical to the regularized least squares solution. Can you identify λ in this solution?

Which prior is implicitly assumed for $\mathbf{w}$ in task 1.1? Explain why the weights explode as the order increases and why we get a flat line if we choose a big λ in task 1.5

Task 1.9

Compute the predictive distribution $p(y|x, \mathbf{x}, \mathbf{y})$ for t given a new value of x and given the training data $\mathbf{y} = (y_1, \ldots, y_N)^{\mathrm{T}}$ and $\mathbf{x} = (x_1, \ldots, x_N)^{\mathrm{T}}$.

Notes:

- Use the following transformations:

$$\begin{aligned}
&\text{given}\\
&p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \boldsymbol{\mu}, \boldsymbol{\Sigma})\\
&p(\mathbf{v}|\mathbf{u}) = \mathcal{N}(\mathbf{v}; \mathbf{A}\mathbf{u} + \mathbf{b}, \mathbf{C})\\
&\text{we can calculate}\\
&p(\mathbf{v}) = \mathcal{N}\left(\mathbf{v}; \mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{C} + \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^{\mathrm{T}}\right)\\
&p(\mathbf{u}|\mathbf{v}) = \mathcal{N}\left(\mathbf{u}; \mathbf{S}\left(\mathbf{A}^{\mathrm{T}}\mathbf{C}^{-1}(\mathbf{v} - \mathbf{b}) + \boldsymbol{\Sigma}^{-1}\mu\right), \mathbf{S}\right)\\
&\text{where}\\
&\mathbf{S} = \left(\boldsymbol{\Sigma}^{-1} + \mathbf{A}^{\mathrm{T}}\mathbf{C}^{-1}\mathbf{A}\right)^{-1}
\end{aligned}$$

- Use the same prior as in task 1.8.
- $p(y|x, \mathbf{x}, \mathbf{y}) = \int p(y|\mathbf{w})p(\mathbf{w}|\mathbf{y})\mathrm{d}\mathbf{w}$

Task 1.10 (Code)

Write a function to compute the mean and variance of $p(y|x, \mathbf{x}, \mathbf{y})$ and use the plot to observe the influence of the model parameters