Solutions to exercises: week 3

Exercise 3.1
(a) A solution can be found by considering the gradient and equating that to 0. Additionally, one needs to realize (though this may not be that trivial) that the function we are minimizing is convex. Maybe we would need a more formal argument for the latter as well...
(b) We can say that we need at least d observations to make sure that the inverse exists. If, however, the subspace spanned by all data contains the origin, we need at least $d+1$ observations. Is this enough for X^TX to be invertible? No, in general we can state that we need the data together with the origin to (linearly) span the whole feature space. Somewhat formally : $(X^TX)^{-1}$ exists if and only if $\dim(\text{span}(\{0, x_1, \ldots, x_N\})) = d$. Note that $d+1$ observations is not sufficient for invertibility, as feature vectors can be linearly dependent.

Exercise 3.2
(a) A solution can be found by considering the gradient and equating that to 0. Additionally, one needs to realize (though this may not be that trivial) that the function we are minimizing is convex. Maybe we would need a more formal argument for the latter as well...
(b) If the data spans the space, we know that $(Z^TZ)^{-1}$ exists and vice versa.
(c) Let's keep it simple and take $X = 1$.

Exercise 3.3
(a) All straight lines but the vertical one that pass through the point (π, e) will minimize the objective function.
(b) One solution is $\{ax + e - \pi a | a \in \mathbb{R}\}$.
(c) Take all 2D planes that go through the points $(1, 1, 1)$ and $(-2, 1, -1)$ except for the one that is perpendicular to the 2D X-plane.
(d) We have a perfect fit, so it's 0!
(e) Take all 2D planes that go through the points $(0, 0, 0)$ and $(1, 1, \pi)$ (except for the one that is perpendicular to the 2D X-plane).
(f) Of course, there is no unique answer to this question. Nevertheless, an argument one can make is that one does not want the solution to be unnecessarily tilted, increasing, or decreasing. In the 1D example this means we would choose a constant solution that goes right through the single training point, i.e., set $a = 0$ and the intercept to e. In the third example, where we forgot about the intercept, out of symmetry considerations, we could decide that if we look in directions perpendicular to the line $x_1 + x_2 = 0$, the regression fit should stay constant. That is, we choose the solution $\frac{\pi}{2}(x_1 + x_2)$, because why would we tilt it more or less to either side of that line? Similarly, but maybe slightly more difficult to see, we would prefer the "flattest" solution $1\frac{1}{2}x_1 - \frac{1}{2}$ for our 2D example with intercept. Interestingly, following this argument, we prefer to pick the solution for which the norm of the (non-intercept) weights is smallest.

Exercise 3.4
(a) We only need to determine the slope, which is easily determined to be $6/14$. So, the function is $6/14x$.
(b) Now we have to invert a matrix! We have $X^TX = (140; 04)$, while $X^TY = (6; 7)$. So, solution is $(140; 04)^{-1}(14; 7) = (6/14; 7/4)\ldots$ and the intercept equals $7/4$.
(c) There are four points, so one needs at least a third-order polynomial to fit these. So the minimum degree is 3. Whether or not there is an intercept present is of no consequence.
Note that this last part is a bit tricky: the answer assumes that we do not need an explicit intercept anymore as it is automatically modelled by the 0th degree monomial anyway. If, however, we mean that such 0th degree are not at all allowed, we need an additional degree of freedom to fit four points, which means we need polynomials of degree 4.

Exercise 3.5
(a) Look up what psd means and realize that for a vector z, z^TX^TXz is always nonnegative, because it equals the inner product of Xz with itself.

(b) $X^T X$ is psd and therefore has only nonnegative eigenvalues (it basically is a covariance matrix). Adding λI makes sure all eigenvalues become positive and therefore this sum of matrices is invertible.
(c) A solution can be found by considering the gradient and equating that to 0. Additionally, one needs to realize (though this may not be that trivial) that the function we are minimizing is convex. Maybe we would need a more formal argument for the latter as well...
(d) The influence of the regularizer becomes smaller and smaller. In the limit, one find the optimal linear least squares fit in which the influence of the regularizer has completely vanished.
(e) In the limit, we get $w = 0$.

Exercise 3.6
(a) One should get the impression that regularization can have quite a beneficial effect.
(b) It may be possible to easily do this in a nifty way using `prcrossval`, but I didn't try. I just wrote some loops that make sure that I run over enough random training sets. For all training set sizes, I found a values around 3 to be fairly optimal.
(c) I did not do the experiment :) but you should find a lower bias and a higher variance for the unregularized solutions.

Exercise 3.7
Simplest solution[?]: generate a single point with a 1D input and a 1D output and show that the solution always goes through this point, irrespective of amount of regularization.

Exercise 3.8
I don't know about you, but I would probably use cross validation for this.

Exercise 3.9
(a) Bias.
(b) variance

Exercise 3.10
If we have a number of samples such that $X^T X$ is invertible, we just get the standard solution back, which is unique to start with and so also is the minimum norm. If $X^T X$ is not invertible, we give the following handwavy "limit" argument. The nonivertibility leads to a whole set of solutions W that all minimize $\|Xw - Y\|^2$, i.e., they all have the same total squared loss. The solution based on the pseudoinverse is obtained by considering the regularized problem $\|Xw - Y\|^2 + \lambda\|w\|^2$, where $\lambda > 0$ shrinks to 0. Among the set of all solutions W, the one with the minimum norm minimizes $\|Xw - Y\|^2 + \lambda\|w\|^2$ for any λ. The limit $\lambda \downarrow 0$ will therefore also give the minimum norm solution.

Exercise 3.11
(a) No, none of the entries ever become zero really. The probability that this happens is 0. In the limit, for λ larer and larger, w should of course shrink to 0 however.
(b) In this setting there will be a finite λ for which at least on of the entries (most often the second of course!) becomes zero. For an even larger λ, also the other entry will become 0.

Exercise 3.12
(b) Of course you expected this behavior! When going up in degree, the variance typically increases, while the bias decreases. When moving to more and more data, the bias remains the same (more or less?), while the variance gets lower and lower.

Exercise 3.14
My quick-and-dirty solution would be two-fold. First, the encoding into months and days is not nice and I would like a more linear kind of time-scale. So, I propose to first transform that into a 1D representation t by something like $t = 30(x_1 - 1) + (x_2 - 1)$. This makes t fairly linear over one year with a minimum of 0 and a maximum of 360. Now, I would expect some periodicity in the signal. So, I actually want to move away from the linear t. I would expect one max and one min temperture in the year, so a first order approximation with a (co)sinus should be a good first

attempt. Based on this, I would use as final 2D input: $(\sin(t), \cos(t))$.

Exercise 3.15
(b) Big error, the fit is basically a constant at 0.
(c) The fit again is basically constant at 0... for all degrees that are not insanely large.
(d) The issues is that `linearr`does not take into account any cross-terms!
(e) The function $y = x_1 x_2$ largely behaves like $y = 50 \sin(x_1) \sin(x_2)$ where the x data is sampled. But for $y = x_1 x_2$ the failure of second and higher order regression should be more apparent as it is actually a second degree polynomial. So, indeed, the issues is that `linearr`does not take into account any cross-terms, i.e., $x_i x_j$, $x_i^2 x_k^5$, $x_i x_j x_k$, etc.

Exercise 3.16
(a) $\binom{m+d-1}{m} = \frac{(m+d-1)!}{(d-1)!m!}$.
(b) If $d > 1$ then yes. The order of polynomial growth for the number of features equals the dimensionality d. You can check this experimentally.
(c) Sure. In bioinformatics one is dealing with gene expression data in which easily $d \leq 10,000$ and so $m = 3$ becomes unmanagable already. Worse even, are image classification problems in which in which one cannot afford to subsample the image. Nowadays one easily has $d > 1,000,000$ and so $m = 2$ already becomesinfeasible.

Exercise 3.17
(a) Fill in $A = I_d$ and $B = X^T$ and work out.
(b) For x an unobserved vector and X the training data, we have $x^T w = x^T (X^T X + \lambda I)^{-1} X^T Y = x^T X^T (X X^T + \lambda I)^{-1} Y$. (Take note of the confusion between row vectors, in X, and the column vector x!) From the last expression, we see that $x^T X^T$ is a vector with inner products between the test vector and all training vectors. Similarly, $X X^T$ is an $N \times N$-matrix containing all inner products between all training feature vectors. Blimey, we have kernelized ridge regression!
(c) $(x^T z + c)^2 = (x^2 z^2 + 2cxz + c^2) = (x^2, \sqrt{2}x, c)(z^2, \sqrt{2}z, c)^T$.
(g) A change in c only changes the scaling of the different feature and we saw already that nonregularized least squares regression is invariant to such scalings (and for very small λ we are basically performing nonregularized regression).

Exercise 3.18
(a) Nadaraya-Watson converges to a flat line at the mean of the output values. Kernelized regression does the same for $\lambda = 0$, but shrinks to 0 for larger and larger λ.
(b) Bias of kernelized regression comes from the basis functions we rely on. Nadaraya-Watson also divides by the overall density in the x values, which results more in an interpolation kind of behavior. For very small width Nadaraya-Watson becomes piecewise constant.
(c) Typically not. But yes, one can enforce it with the right parameter choice.

Exercise 3.19
(a) We have already shown that $w = (X^T X)^+ X^T Y$ solves the regression task. Realizing that $(X^T X)^+ X^T Y = 2(X^T X)^+ N \frac{1}{2N} X^T Y = 2(\frac{1}{N} X^T X)^+ \frac{1}{2N} X^T Y$ and then working out the two components gets you to the solution.
(b) What we need to show is that the regressor w learned before the transformation applied to an untransformed x gives the same output as the regressor trained on the transformed data applied to the transformed x.

Exercise 3.20
(a) I'm not going to be precise (intercept yes/no etc.). So, one way to describe it is that $H = \mathbb{R}^d$ and, with $D = (X, Y)$, the loss is $\|Xh - Y\|^2$. Finally, the regularization term is given by $\lambda \|h\|^2$.
(b) The hypothesis class is the set of all Gaussian distributions (in $\mathbb{R}^d$, say). Of course, you should be able to write more explicitly how that set looks like... With $D = X$, the loss is the negative(!) (log-)likelihood: $\prod_{i=1} h(x_i)$. We don't have a regularization term.

Exercise 3.21
(a) With no intercept, we have $\ell(x, y|h) = (h(x) - y)^2 = (w^T x - y)^2$ if we assume that h is

parametrized by $w \in \mathbb{R}^d$.
(b) Given that $h \in H$ is a probabilistic model and (x, y) is a point from the training data, we could define the loss as $-\log h(x, y)$. Often, we make explicit that H can be parametrized (say through some θ) and we would consider the corresponding loss $-\log h(x, y|\theta)$.
(c) One example is AUC.
(d) I would go for 1NN. Or better 3NN.

Exercise 3.22
(a) Every term in the objective function can be written as $\log_2(1 + e^{-y_i x_i^T w})$, so we have $v(a) = \log_2(1+e^{-a})$ (or some equivalent expression of course). Note that this also shows that the definition of a classifier in terms of its hypothesis class, its loss, and its regularization function is not unique: there are multiple formulations that lead to the same classifier.
(b) $v(a) = (1 - a)^2$.
(c) This may not be obvious. Answer: $v(a) = \max(0, 1 - a)$.
(d) I don't think so. But if you have an idea, I would be happy to discuss (ML).

Exercise 3.23
(a) Let us use the notation $N(a|m, s)$ for the normal distribution with mean m and variance s^2 for the variable a, then we simply have $p(x, y|w) = N(x|\nu, \tau)N(y|x^T w + w_0, \sigma)$.
(b) We just get the standard linear least squares solution back.
(c) When taking the derivative of the log-likelihood to w, the θ disappears from the equations and vice versa.
(d) We already determined $\hat{w}$ and $\hat{w}_0$ in the first part of this exercise. In addition, $\hat{\sigma} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - x_i^T \hat{w} - \hat{w}_0)^2}$.