Problem Set 3

Problem 1 (Smell-Tests for Classification Model)

Please take some time (30 minutes is likely enough) to briefly read the paper "Automated inference on criminality using face images," available at this link.

Then, please write a paragraph in which you respond to the following questions:

1. What is the *task* that the authors address? What is the target variable they are trying to predict, and what are the features being used to make the prediction?
2. What is the main *finding* or *achievement* that the authors present?
3. If a major newspaper wrote an article about this research paper, what would the headline be?
4. If you read that headline, what would your reaction be?
5. Does anything smell funny to you about the authors' methods?

Problem 2 (Thresholds in Decision-Making)

Your so-called friend has convinced you to play the following ("fun!") game.

1. Your friend will flip a biased coin with probability of heads equal to q. As the coin spins in the air, you need to predict whether the coin lands heads or tails.
   - Your friend tells you the value of q before the coin flip.
2. You receive a *payoff* depending on your prediction and the outcome of the coin flip, according to the following table:

Prediction	Outcome	Payoff
Heads	Heads	+1
Tails	Tails	+1
Heads	Tails	$-a$
Tails	Heads	$-b$

Here, a and b are positive numbers that represent the cost of calling the flip wrong.

Part A

Suppose you predict heads. Write down a formula for your expected payoff in terms of q, a, and b. Write down a similar formula for the expected payoff if you predict tails.

Part B

Suppose that you decide to follow a policy: you pick $t \in [0, 1]$ in advance. Then, when your friend tells you the value of q, you predict heads if $q \geq t$ and tails otherwise. What is the value of t that maximizes your expected payoff?

Problem 3 (Sigmoid Gradient)

Part A

Let $\sigma : \mathbb{R} \to (0, 1)$ be the *logistic sigmoid* function, defined by $\sigma(z) = \frac{1}{1+e^{-z}}$.

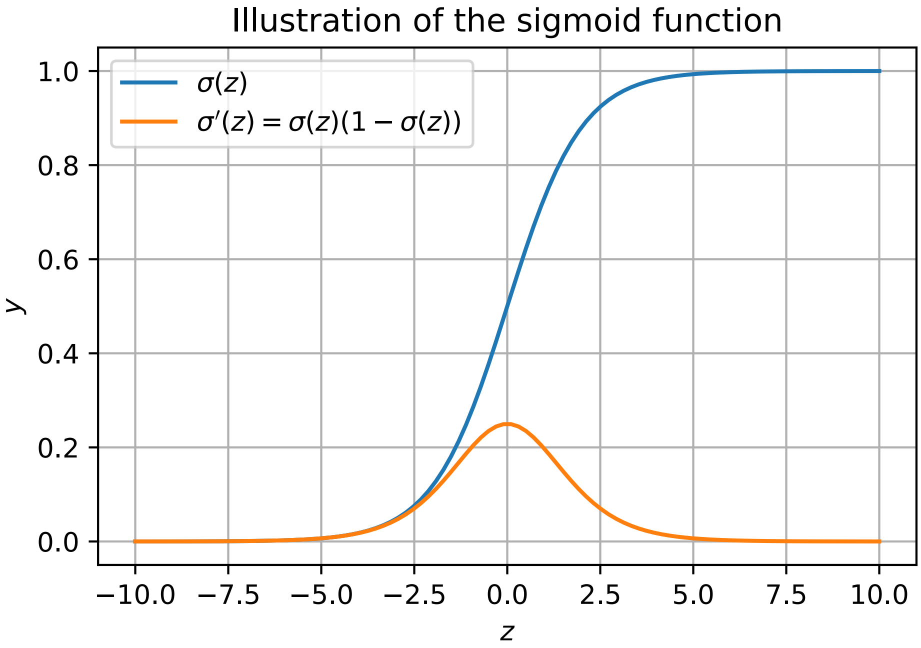


Figure 1: Plot of the logistic sigmoid function and its derivative.

Prove by calculation that for any $z \in \mathbb{R}$, we have

$$\frac{d}{dz}\sigma(z) = \sigma(z)(1 - \sigma(z)).$$

Part B

Let $q : \mathbb{R}^d \to (0, 1)$ be the function defined by $q(\mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x})$ for some $\mathbf{w} \in \mathbb{R}^d$. Compute the gradient $\nabla q(\mathbf{x})$, with the partial derivatives taken with respect to the entries of $\mathbf{w}$. Please express your final answer in *vectorized* form, i.e. as a function of $\mathbf{w}$, $\mathbf{x}$, and $q(\mathbf{x})$, without any summations or explicit indices.

Hint: Start by computing $\partial q(\mathbf{x}) / \partial w_i$ for an arbitrary $i \in \{1, 2, ..., d\}$, and then use the result to write down the full gradient.

Problem 4 (Convex Functions)

A twice-differentiable function $g : \mathbb{R} \to \mathbb{R}$ is (globally) *strictly convex* if and only if its second derivative is everywhere positive: $g''(x) > 0$ for all $x \in \mathbb{R}$ (this fact is often called the *second derivative test.*). An important feature of globally strictly convex functions is that they have only one critical point (point where $g'(x) = 0$), and that critical point is a global minimum. Strictly convex functions are extremely helpful in optimization contexts – they are guaranteed to have a unique global minimum, and gradient-based optimization algorithms are guaranteed to find that minimum.

It turns out that both the mean-squared error for linear regression and the cross-entropy loss for logistic regression are globally strictly convex functions of the the signal $s = \mathbf{w}^\top \mathbf{x}$ (this is sufficient to guarantee that the loss is globally strictly convex as a function of the model parameters $\mathbf{w}$). In this problem, you'll prove the key step in showing that these functions are globally strictly convex, which is to show that they are globally strictly convex as functions of the signal s.

Note: *The main additional fact needed to prove that the MSE and cross-entropy is convex as a function of* $\mathbf{w}$ *(rather than* s*) is to note that* $s = \mathbf{w}^\top \mathbf{x}$ *is a linear function of* $\mathbf{w}$*, and that the composition of a convex function with a linear function is also strictly convex, this time in the sense of multivariate functions.*

Part A

Use the second derivative test to show that the function

$$g(s) = (y - s)^2$$

is globally strictly convex as a function of s.

Part B

Use the second derivative test to show that the function

$$g(s) = -y \log(\sigma(s)) - (1 - y) \log(1 - \sigma(s))$$

is globally strictly convex as a function of s, for $s \in \mathbb{R}$ and $y \in \{0, 1\}$, where $\sigma(s) = \frac{1}{1+e^{-s}}$ is the logistic sigmoid function.

Problem 5 (Playing with Penguins)

Part A

First, fire up a blank notebook in Google Colab. Then, follow the lecture notes to download the Palmer Penguins data set, restrict the columns to `Culmen Length (mm)`, `Culmen Depth (mm)`, and `Species`, and convert the `Species` column to dummy variables. Perform a train-test split.

Your result should be a training set `X_train` of shape $(n_{\text{train}}, 2)$, a training label tensor `y_train` of shape $(n_{\text{train}}, 3)$, a test set `X_test` of shape $(n_{\text{test}}, 2)$, and a test label vector `y_test` of shape $(n_{\text{test}}, 3)$.

You are welcome to copy (with attribution) as much of the lecture code as you find convenient.

Part B

Freely copying (with attribution), include in your notebook a complete working implementation of multinomial logistic regression. Please produce a version of the righthand plot of Fig. 9.3, which relies on the `plot_decision_regions` function defined in the lecture notes. You should include the decision regions, the training data points, a legend, and the axis labels.

Part C

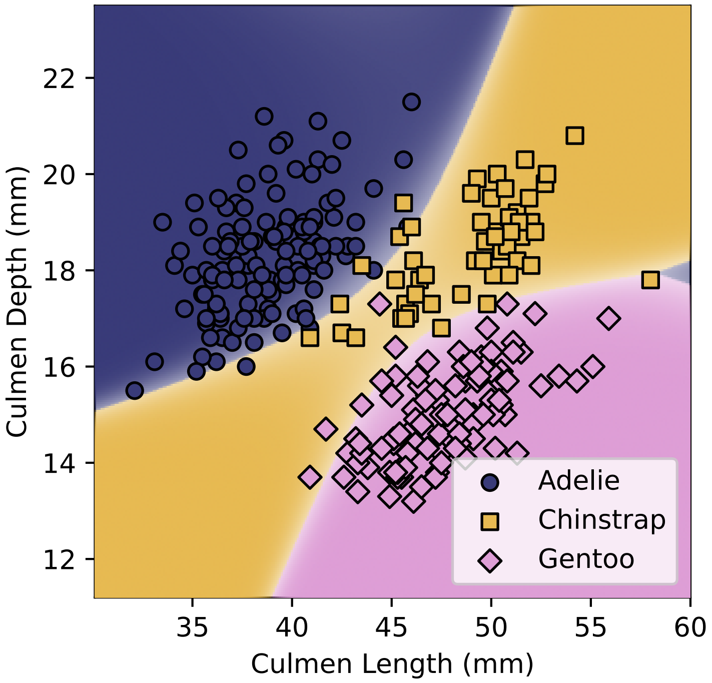
Now, *add nonlinear feature maps* to your model. You're free to use any nonlinear feature map you like. Train your model again and produce a plot of the decision regions as above. You should now observe nonlinear decision boundaries, although the extent of nonlinearity may be somewhat faint. The only requirement here is that the nonlinearity of the decision boundaries be visible by eye.

Hints:

- To produce the plot, I found it helpful to add a `feature_map` argument to the `plot_decision_regions` function.
- You may need to experiment with the learning rate and number of epochs to get good results.
- A feature map which I found gave interesting results was a modified quadratic feature map of the form

$$\phi(\mathbf{x}) = \begin{bmatrix} x_1 \\ x_2 \\ (x_1 - \bar{x}_1)^2 \\ (x_2 - \bar{x}_2)^2 \\ (x_1 - \bar{x}_1)(x_2 - \bar{x}_2) \end{bmatrix},$$

where $\bar{x}_1$ and $\bar{x}_2$ are the means of the first and second features, respectively. Using this feature map, I was able to obtain decision regions that look like this:



Part D

Finally, evaluate your accuracy on the test data set. To receive full credit on this problem, it's sufficient that your decision boundaries are visibly nonlinear and that your test accuracy is above 90%.

Problem 6 (Why the logistic sigmoid?)

Consider the following generative model of 1D binary classification data. In this model, we generate *both* the class labels $y \in \{0, 1\}$ *and* the feature values $x_i \in \mathbb{R}$ according to a probabilistic process. For each training example i:

1. First, we flip a biased coin with probability of heads equal to p. If the coin comes up heads, we assign the class label $y_i = 1$, and if the coin comes up tails, we assign the class label $y_i = 0$.
2. Next, we draw the feature value x_i from a class-conditional Gaussian probability distribution. This means that

$$x_i \sim \mathcal{N}(\mu_i, \sigma^2),$$

where μ_i is the mean of the Gaussian distribution for class y_i, and σ^2 is the variance (which we assume to be the same for both classes). We can write the probability density function for x_i given y_i as:

$$p_{X \mid Y}(x_i \mid y_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu_i)^2}{2\sigma^2}\right).$$

Now suppose that we are given the feature values x_i, as well as the parameters of the generative model (i.e., p, μ_0, μ_1, and σ^2). To determine the *posterior probability* that point y_i has class 1, we need to use Bayes' rule:

$$\mathbb{P}(y_i = 1 \mid x_i) = \frac{p_{X \mid Y}(x_i \mid y_i = 1)\mathbb{P}(y_i = 1)}{p_{X \mid Y}(x_i \mid y_i = 0)\mathbb{P}(y_i = 0) + p_{X \mid Y}(x_i \mid y_i = 1)\mathbb{P}(y_i = 1)}$$

Prove that the posterior probability $\mathbb{P}(y_i = 1 \mid x_i)$ can be expressed in the form of a logistic function, i.e.,

$$\mathbb{P}(y_i = 1 \mid x_i) = \frac{1}{1 + \exp(-(\alpha x_i + \beta))} = \sigma(\alpha x_i + \beta),$$

where α and β are constants that depend on p, μ_0, μ_1, and σ^2, and σ is the logistic sigmoid (not to be confused with the variance parameter σ^2). Please give formulas for α and β in terms of these parameters.

Hint: This sounds scary, but all you need to do is plug in the Gaussian density functions into the Bayes' rule formula above and do some algebra to simplify the result into the desired form. A possibly-helpful algebra formula is:

$$\frac{e^a}{e^a + e^b} = \frac{1}{1 + e^{b-a}} = \sigma(b - a).$$

Note: This problem gives one justification for the appearance of the logistic sigmoid σ in logistic regression: if we assume that the data came from a process like the one described in the problem, then the logistic sigmoid gives the correct estimate of the probability of a given class membership given the value of the feature.