Machine Learning
Exercises

Christian Rossi

Academic Year 2023-2024

Abstract

The course will cover several topics, starting with an introduction to basic concepts. Learning theory will be explored, including the bias-variance tradeoff, Union and Chernoff/Hoeffding bounds, VC dimension, worst-case (online) learning, and practical advice on using learning algorithms effectively.

In supervised learning, key areas of focus include the supervised learning setup, LMS, logistic regression, perceptron, the exponential family, and kernel methods such as Radial Basis Networks, Gaussian Processes, and Support Vector Machines. Additionally, topics like model selection, feature selection, ensemble methods (e.g., bagging and boosting), and strategies for evaluating and debugging learning algorithms will be addressed.

The course will also delve into reinforcement learning and control, examining Markov Decision Processes (MDPs), Bellman equations, value iteration, policy iteration, TD, SARSA, Q-learning, value function approximation, policy search, REINFORCE, POMDPs, and the Multi-Armed Bandit problem.

Contents

CHAPTER 1

Linear regression

1.1 Exercise 1

Given the relationship:

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

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.
2. Independent variables.
3. Features.
4. Model.

Which kind of problem do you think it is trying to solve?

Solution

In the proposed relationship we have:

1. The response (or target or output) is the amount of sales S.
2. The independent variables (or input) are TV, R and N
3. The features (or input) are TV, R and N.
4. The model is identified by the function $f(\cdot)$.

Since the amount of sales S is a continuous and ordered variable, we are trying to solve a regression problem (supervised learning).

1.2 Exercise 2

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.

Solution

1. False: it rarely happens that the problem we are modeling has linear characteristics.
2. True: it is true that this model is easy to interpret and can be extended to also consider nonlinear relationships among variables, e.g., using basis functions.
3. True: since we are able only to minimize the discrepancy between the considered function and the real one, and we can not reduce the variance introduced by noise, the use of linear model might be a better choice with respect to more complex ones since they are usually prone to overfitting, i.e., they try to model also the noise of the considered process.
4. True: they are easy to interpret and might give suggestions on more sophisticated techniques which can be used to tackle specific problems.
5. True/False: for some loss functions we have a closed form solution for linear model (LS method), thus we can guarantee that we are able to find the parameters minimizing the loss function effectively. That is not always true and depends also on the loss function we want to minimize.

1.3 Exercise 3

Consider a linear regression with input x, target y 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?

Motivate your answers.

Solution

The original formulation is:

$$y = \theta^* x$$

1. In this scenario, the formulation simplifies to:

   $$y = \theta_1 x + \theta_2 2x$$

   As the two variables are dependent, it can alternatively be expressed as:

   $$y = \underbrace{(\theta_1 + 2\theta_2)}_{\theta^*} x$$

   Thus, yielding the original formulation:

   $$y = \theta^* x$$

   Moreover, for computing the closed-form optimization, the formula employed is:

   $$\omega = \left(x^T x\right)^{-1} xt$$

   However, in this particular case, the matrix x takes the form:

   $$x = \begin{bmatrix} x_1 & 2x_1 \\ x_2 & 2x_2 \\ \vdots & \vdots \\ x_n & 2x_n \end{bmatrix}$$

   Hence, $x^T x$ becomes singular, rendering the previous formula inapplicable.

   In general, if x lacks full rank, then $x^T x$ becomes singular, and its inverse cannot be computed.

2. The parameter we get have a high variance, since we have an infinite number of couples of parameters minimizing the loss of the samples in the considered problem. Indeed, if the parameters of the two inputs are w_1 and w_2 we would have that the true relationship would be:

   $$y = \theta_1 x + \theta_2 2x = (\theta_1 + 2\theta_2)\, x$$

   Which can be satisfied by an infinite number of solutions.

3. In this case, the use of ridge regression is able to partially cope with the influence of using highly linearly correlated features. Another viable option is to remove the variables which are linearly dependent, for instance by checking if they have correlation equal to 1 or -1.

4. In this case we do not have a badly conditioned matrix since x and x^2 are not linearly dependent and the corresponding design matrix would not be ill-conditioned. As a result, we can find a closed form optimization.

1.4 Exercise 4

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

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.

Solution

Since the matrix $\left(\Phi^T\Phi + \lambda I\right)$ is definite positive and its eigenvalues should all be greater than $\lambda = 10^{-5}$, we have:

1. Not plausible: one eigenvalue is smaller than 10^{-5}.
2. Not plausible: one eigenvalue is negative.
3. Plausible: all positive and greater than 10^{-5}.
4. Plausible: all positive and greater than 10^{-5}.

CHAPTER 2

Classification

2.1 Exercise one

Classify the following variable as quantitative or qualitative:

1. Height.
2. Age.
3. Speed.
4. Color.

Provide a technique for transforming qualitative data into quantitative format without imposing additional organization on the data.

Solution

The classification is as follows:

1. Quantitative variable: heights can be ordered and typically belong to a bounded continuous set.
2. Quantitative variable: values are ordered natural numbers.
3. Quantitative variable: takes real number values.
4. Qualitative variable: since there's no inherent order among colors, one-hot encoding can be employed without imposing additional structure on the data.

For a set of all possible colors, $\mathcal{C} = \{c_1, \ldots, c_p\}$ one-hot encoding creates a binary variable $z_i \in \{0, 1\}$ for each color c_i. This variable equals one when the color is c_i. Thus, color c_i is represented by a binary vector $z_i = \begin{bmatrix} z_1 & \cdots & z_n \end{bmatrix}^T$, where $z_i = 1$ and all other $z_j = 0$ for $j \neq i$. This method introduces p new variables without further structuring the data. Notably, any two vectors $zi \neq z_j$ are equally distant under reasonable metrics like Euclidean distance.

It's important to note that assigning a quantitative variable i to each color would introduce additional structure to the data, which should generally be avoided. While this approach requires only one variable instead of p, it imposes an ordering among the colors. Moreover, it results in color c_i being closer to color $c_i + 1$ than to color $c_i + 2$ in terms of Euclidean distance.

2.2 Exercise two

Consider a dataset comprising workers' attributes such as the number of hours spent working (x_1), the number of completed projects (x_2), and whether they received a bonus (t). After applying logistic regression, we obtain the following coefficients: $w_0 = -6$, $w_1 = 0.05$, and $w_2 = 1$.

1. Determine the likelihood of a worker receiving a bonus given that they worked for 40 hours and completed 3.5 projects.
2. Calculate the number of hours a worker needs to work to have a 50% chance of receiving a bonus.
3. Discuss whether values of z in $\sigma(z)$ lower than -6 are meaningful in this context, and provide reasoning.

Solution

1. The logistic model yields the probability of receiving a bonus as its output, expressed by:

$$\mathrm{P}(t = 1|\mathbf{x}) = \sigma(w_0 + w_1 x_1 + w_2 x_2)$$

   Given $x_1 = 40$ and $x_2 = 3.5$:

$$\mathrm{P}(t = 1|\mathbf{x}) = \sigma(-6 + 0.05 \cdot 40 + 1 \cdot 3.5) = \sigma(-0.5) = 0.3775$$

2. To ascertain the probability of receiving a bonus with a confidence level $\alpha\%$, we need to invert the sigmoidal function. However, in this instance, a 50% chance corresponds to the sigmoidal argument being zero. Hence:

$$w_0 + w_1\hat{x} + w_2 x_2 = 0 \rightarrow -6 + 0.05\hat{x} + 3.5 = 0 \rightarrow \hat{x} = 50$$

3. Considering that all the variables under consideration are positive definite, it is reasonable to regard predictions with values greater than -6 as meaningful.

2.3 Exercise three

Consider a binary classifier trained on a dataset comprising $N = 100$ samples.

1. Given a precision of 0.25 and an F1 score of 0.4, compute the recall.
2. Additionally, with an accuracy of 0.85, calculate the complete confusion matrix.
3. Under what circumstances is accuracy not a dependable metric for evaluating the model's quality?

Solution

1. We have:

$$\text{F1} = \frac{2 \cdot \text{Pre} \cdot \text{Rec}}{\text{Pre} + \text{Rec}} = 0.4$$

$$\text{Pre} = \frac{TP}{TP + FP} = 0.25$$

We seek to find:

$$\text{Rec} = \frac{TP}{TP + FN}$$

Substituting, we obtain:

$$\frac{2 \cdot 0.25 \cdot \text{Rec}}{0.25 + \text{Rec}} = 0.4 \rightarrow \text{Rec} = 1$$

2. Given:

$$\text{F1} = \frac{2 \cdot \text{Pre} \cdot \text{Rec}}{\text{Pre} + \text{Rec}} = 0.4$$

$$\text{Pre} = \frac{TP}{TP + FP} = 0.25$$

$$\text{Rec} = \frac{TP}{TP + FN} = 1$$

$$\text{Acc} = \frac{TP + TN}{N} = 0.85$$

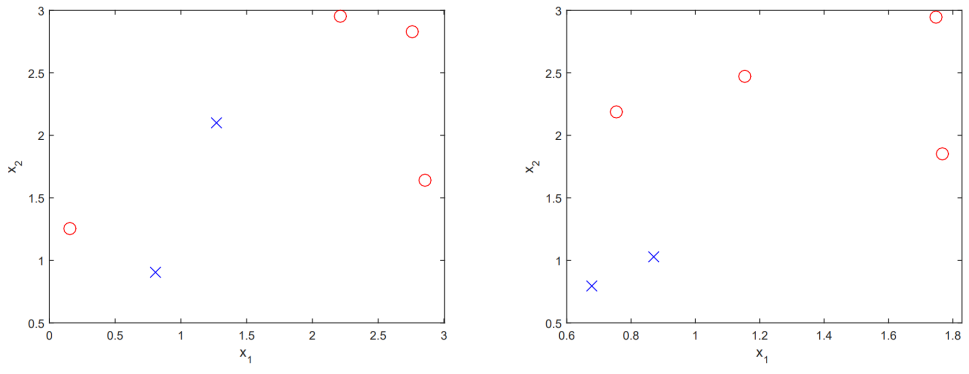
We infer that $FN = 0$ since the recall is unity. Then, from the other formulas:

$$\begin{cases} \text{TP} + \text{TN} = 85 \\ \text{FN} = 0 \\ \text{TP} = 0.25(\text{TP} + \text{FP}) \\ \text{TP} + \text{TN} + \text{FN} + \text{FP} = 100 \end{cases} \rightarrow \begin{cases} \text{TP} = 5 \\ \text{FN} = 0 \\ \text{FP} = 15 \\ \text{TN} = 80 \end{cases}$$

3. Accuracy is not a reliable indicator of the model's quality primarily under two conditions: when the dataset is imbalanced, and when the consequences of misclassifying positive-class samples differ from misclassifying negative-class samples.

2.4 Exercise four

Consider the provided datasets:



Now, let's analyze whether the learning procedure terminates and the number of steps required for convergence using the online stochastic gradient descent algorithm to train a perceptron.

Solution

The perceptron learning algorithm converges if there exists a linear separation hyperplane. In such a scenario, the classification error can be reduced to zero. If no linear separation exists, the optimization process doesn't halt. The convergence rate isn't assured since it relies on the initial parameterization and the sequence of points used for training. Nonetheless, convergence does occur within a finite number of steps.

In the first dataset (left), convergence isn't guaranteed. However, in the second dataset (right), the online stochastic gradient descent will ultimately converge within a finite number of steps.

Considering that the loss function for logistic regression is convex, the online learning process converges to the global optimum asymptotically, irrespective of the dataset provided.

CHAPTER 3

Bias-variance tradeoff

3.1 Exercise one

As you're fitting a linear model to your dataset, you contemplate transitioning to a quadratic model, incorporating quadratic features $\varphi(x) = \begin{bmatrix} 1 & x & x^2 \end{bmatrix}$. Considering this change, which of the following statements is most likely true:

1. Employing the quadratic model will reduce your reducible error.
2. Employing the quadratic model will decrease the bias in your model.
3. Employing the quadratic model will reduce the variance in your model.
4. Employing the quadratic model will decrease your reducible error.

Provide motivations to your answers.

Solution

1. False: changing the model doesn't directly impact the irreducible error, as it's inherent to the problem itself and can't be mitigated by model choice.
2. True: expanding the model to include quadratic features increases its flexibility, making it better able to capture complex relationships in the data. Therefore, it's likely to reduce bias or, at the very least, not increase it.
3. False: introducing more complex features typically leads to increased variance as the model becomes more sensitive to fluctuations in the training data.
4. Partially true: whether using the quadratic model decreases the reducible error depends on the balance between bias reduction and variance increase. If the quadratic model effectively reduces bias without excessively inflating variance, it can lead to a more accurate model and decrease reducible error. However, if the increase in variance outweighs the reduction in bias, the overall error may increase. Therefore, the statement could be true or false depending on the specific circumstances.

3.2 Exercise two

We determine the regression coefficients in a linear regression model by minimizing ridge regression for a specific value of λ. For each of the following, elucidate the trend of the elements as we increment λ from 0 (e.g., remains constant, increases, decreases, increases and then decreases):

1. The training RSS.
2. The test RSS.
3. The variance.
4. The squared bias.
5. The irreducible error.

Solution

1. Increases: as λ increases, simpler models are favored, leading to a decrease in flexibility and an inability to fit the training data precisely. Consequently, the training RSS will steadily increase.
2. Decreases and then increases. Initially, as λ increases, the test RSS improves due to a reduction in overfitting on the training data. However, beyond a certain point, overly simplistic models fail to capture the true underlying patterns, causing the test RSS to increase.
3. Decreases: increasing λ forces the use of simpler models, which inherently reduces the variability of the fits across different datasets.
4. Increases: with higher λ values, simpler models are employed, likely resulting in larger squared bias as these models may fail to capture the true underlying relationships adequately.
5. Remains constant: increasing λ does not affect the irreducible error since it is independent of the model's complexity and solely depends on the inherent noise in the data.

3.3 Exercise three

What methods would you employ to assess the efficacy of various models given the following scenarios:

1. Limited dataset size with straightforward models.
2. Limited dataset size with intricate models.
3. Extensive dataset with basic models.
4. Extensive dataset with access to parallel computing capabilities for training.

Justify you choices.

Solution

1. Leave-one-out cross-validation (LOO): when dealing with a small dataset and simple models, LOO is a viable option as it doesn't pose significant computational complexity. This method offers a nearly unbiased estimation of the test error.

2. Akaike information criterion (AIC) with Adjustment Techniques: with a smaller dataset, training might lead to overfitting, rendering traditional methods ineffective. AIC, with its adjustment techniques, can help mitigate overfitting concerns. However, for complex models, LOO may still be impractical due to computational constraints.

3. Cross-validation (CV): cross-validation is suitable for obtaining stable estimates to select the best model, particularly when Leave-One-Out is infeasible due to computational complexity. It balances the need for reliable estimates with computational efficiency.

4. Parallelized leave-one-out (LOO): in scenarios where parallel computing resources are available, and a large dataset is being utilized, parallelizing LOO can significantly reduce computation time. By concurrently training multiple models, the time required for LOO can be reduced by a factor equal to the number of parallel processes running simultaneously.

CHAPTER 4

Model selection

4.1 Exercise one

Consider the following statement regarding PCA and tell if they are true or false. Provide motivation for your answers.

1. Even if all the input features are on very similar scales, we should still perform mean normalization (so that each feature has zero mean) before running PCA.

2. Given only scores t_i and the loadings W, there is no way to reconstruct any reasonable approximation to x_i.

3. Given input data $x_i \in \mathbb{R}^d$, it makes sense to run PCA only with values of k that satisfy $k \leq d$.

4. PCA is susceptible to local optima, thus trying multiple random initializations may help.

Solution

1. True: since the principal components are identifying the directions where the most of the variance of the data is present, where the directions is defined as a vector with tail in the origin, we should remove the mean values for each component in order to identify correctly these directions.

2. False: by applying again the loadings matrix W to the scores t_i, thanks to the orthogonality property of W, we are able to reconstruct perfectly the original mean normalized vectors. If we want to reconstruct the original vectors we should also store the mean values for each dimension.

3. True: running it with $k = d$ is possible but usually not helpful and $k > d$ does not make sense.

4. False: there is no source of randomization and no initialization point in the algorithm to perform PCA.

4.2 Exercise two

State whether the following claims about Bagging and Boosting are true or false, motivating your answers:

1. Since Boosting and Bagging are ensemble methods, they can be both parallelized.
2. Bagging should be applied with weak learners.
3. The central idea of Boosting consists in using bootstrapping.
4. It is not a good idea to use Boosting with a deep neural network as a base learner.

Solution

1. False, only Bagging can be parallelized, since training is done on different datasets, while Boosting is sequential by nature.
2. False, weak learners are good candidate for Boosting, since they have low variance. Typically, one uses instead bagging when more complex and unstable learners are needed, to reduce their variance.
3. False, bootstrapping is used in bagging, whose name derived indeed from boosting aggregation.
4. True, it is not a good idea to do that, since deep neural networks are very complex predictor, which can have large variance. Therefore, you may not succeed in lowering bias without increasing variance. Moreover, since you need to train the network multiple times, the procedure may require a lot of time.

4.3 Exercise three

Answer to the following questions about the bias-variance decomposition, model selection, and related topics. Motivate your answers.

1. If your linear regression model underfits the training data (i.e., the model is not complex enough to explain the data), would you apply PCA to compute a more suitable feature space for your model?
2. If solving a regression problem, the design matrix $X^T X$, is singular, would you apply PCA to solve this issue?
3. Assuming a classifier fits very well the training data but underperforms on the validation set, would you apply Bagging or Boosting to improve it?
4. Assuming that you trained a classifier with a K-fold cross-validation and it consistently has poor performances both on training and on validation folds, would you apply Bagging or Boosting to improve it?
5. You applied ridge regression to train a linear model using a rather large regularization coefficient, would you think that bagging would improve your model?

6. You have been asked to implement a feature selection process on a system with very limited computational resources. Would you opt for a filter approach or for a wrapper approach?

7. You have been asked to implement a feature selection process to improve as much as possible the performance of your model. Would you opt for a filter approach or for a wrapper approach?

8. You need to train a linear regression model using as input the readings of several sensors. Assuming that you know that some of these sensors might be faulty (i.e., resulting in meaningless readings), which linear regression approach would you use to train your model?

9. A linear regression model, computed using ordinary least squares, has a validation error that is much larger than training error. Assuming that you do not want to change neither the input features nor the kind of model, what would you do to improve it?

10. If you have to choose among a few models knowing only the training error (assuming you cannot retrain them or evaluate them on a different dataset), what would you do?

Solution

1. No, because if linear regression does not fit training data (underfitting), the features computed with PCA will not solve the problem as they are linear combinations of input variables.

2. Yes, applying PCA and selecting the top K components we will allow to avoid co-linearity in the resulting feature space.

3. Bagging, because it might reduce the variance of the model. In contrast, boosting allows to reduce bias without increasing (significantly) the variance (however, in this example we have a low bias and high variance).

4. Boosting, because it can successfully reduce bias of a stable learner without increasing the variance which seems to be the problem of the learner in this case.

5. No, because ridge regression with large regularization coefficient will be very stable (limited variance) and this would not allow to exploit significantly bagging.

6. Filter, because a wrapper approach involves solving an optimization problem that requires training several models. In contrast, filter approaches only require to compute statistics on the features.

7. Wrapper, because filter approaches assume features are independent and might not find the best subset.

8. Lasso, because it implicitly performs a feature selection that will get rid of the faulty sensors.

9. Regularization can help me improve the performance by reducing the variance. In particular ridge regression would be the most obvious choice. Another solution, if viable, is to increase the number of samples used for training.

10. Adjusted complexity matrix can be used to correct the training error taking into account also the model complexity.

CHAPTER 5

Kernel methods

5.1 Exercise one

Tell if the following functions are valid kernels. Motivate your answers. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$:

1. $k_1(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T\mathbf{y} + \mathbf{x}^T\mathbf{1} + \mathbf{y}^T\mathbf{1} + \mathbf{d}$, where $\mathbf{1} \in \mathbb{R}^d$ is the vector of all ones.
2. $k_2(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T\mathbf{y} - \|\mathbf{x}\|^2$.
3. $k_3(\mathbf{x}, \mathbf{y}) = k_1(\cos(\mathbf{x}), \cos(\mathbf{y}))^3$, where the $\cos(\cdot)$ function is applied element-wise.
4. $k_4(\mathbf{x}, \mathbf{y}) = e^{(k_2(\mathbf{x},\mathbf{y})+k_2(\mathbf{x},\mathbf{y}))}$.

Solution

1. True: by definition, selecting $\phi(\mathbf{x}) = \mathbf{x} + \mathbf{1}$, we have $k_1(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T\phi(\mathbf{y})$.
2. False: $k_2(\mathbf{x}, \mathbf{y})$ is not symmetric.
3. True: since we are considering the same transformation $\cos(\cdot)$ applied to both arguments, then $(\cdot)^3$ is a polynomial transformation with non-negative coefficients, and k_1 is a kernel.
4. True: with simple computations, we obtain:

$$k_4(\mathbf{x}, \mathbf{y}) = e^{-k_2(\mathbf{x},\mathbf{y})+k_2(\mathbf{y},\mathbf{x})} = e^{2\mathbf{x}^T\mathbf{y}-\|\mathbf{x}\|^2-\|\mathbf{y}\|^2} = e^{-\|\mathbf{x}-\mathbf{y}\|^2}$$

That is the Gaussian kernel with $\sigma = \frac{1}{2}$.

5.2 Exercise two

Comment on the following statements about Gaussian Processes (GP). Assume to have a dataset generated from a GP $\mathcal{D} = (x_i, y_i)_{i=1}^N$. Motivate your answers.

1. GPs are parametric methods.
2. The computation of the estimates of the variance of the GP $\hat{\sigma}^2(x)$ corresponding to the input x provided by $\mathcal{D}$ does not require the knowledge of the samples output $(y_1, \ldots, y_N)$.

3. In the neighborhood of the input points $(x_1, \dots, x_N)$, we observed the variance of the GP gets smaller and smaller as we collect more samples.
4. The complexity of the computation of the estimates of the mean $\hat{\mu}(x)$ and variance $\hat{\sigma}^2(x)$ scales as N^3, i.e., cubically with the number of samples N.

Solution

1. False: they require to store the gram matrix whose dimension depends on the number of samples.
2. True: it requires only the gram matrix and the computation of the kernel on the new point.
3. True: the uncertainty we have around the sampled points decreases as we get more and more samples.
4. True: indeed, it requires the inversion of the gram matrix which has N^3 computational cost.

5.3 Exercise three

Consider the linear two-class SVM classifier defined by the parameters $w = \begin{bmatrix} 2 & 1 \end{bmatrix}$, $b = 1$. Answer the following questions providing adequate motivations.

1. Is the point $x_1 = \begin{bmatrix} -2 & 4 \end{bmatrix}$ a support vector?
2. Give an example of a point which is on the boundary of the SVM.
3. How the point $x_2 = \begin{bmatrix} 3 & -1 \end{bmatrix}$ is classified according to the trained SVM?
4. Assume to collect a new sample $x_3 = \begin{bmatrix} -1 & 2 \end{bmatrix}$ in the negative class, do we need to retrain the SVM?

Solution

1. A point is a support vector if $\left|w^T x + b\right| \leq 1$, thus:

$$\left|w^T x + b\right| = |-4 + 4 + 1| = 1$$

   Meaning that x_1 is a support vector.

2. A point on the boundary has to satisfy $w^T x + b = 0$ thus by considering $x_{11} = 0$:

$$2 \cdot 0 + 1 \cdot x_{22} + 1 = 0 \rightarrow x_{22} = -1$$

   Thus $x = \begin{bmatrix} 0 & -1 \end{bmatrix}$ is on the boundary.

3. A point is classified either in the positive class or in the negative one if $w^T x + b$ is positive or negative, respectively, thus:

$$w^T x_2 + b = 2 \cdot 3 - 1 \cdot 1 + 1 = 6$$

   Which means that the point it classified in the positive class.

4. The point is misclassified by the current model, as it is

$$w^T x_3 + b = -2 + 2 + 1 = 1$$

Which means that x_3 would be a support vector, thus we need to retrain the model.

CHAPTER 6

Learning theory

6.1 Exercise one

1. Show that the VC dimension of an axis aligned rectangle is 4.
2. Show that the VC dimension of a linear classifier in 2D is 3.
3. Show that the VC dimension of a triangle in the plane is at least 7.
4. Show that the VC dimension of a 2D stump, i.e., use either a single horizontal or a single vertical line in 2D to separate points in a plane, is 3.

Solution

1. Consider 4 points. It is possible to show by enumeration that all the possible labeling are shattered by the rectangle. Consider 5 points. Consider the set of points with maximum and minimum x coordinate and maximum and minimum y coordinates. If all the points are on the rectangle, we consider the labeling which assign alternate labels to the points if you follow the rectangle perimeter. Otherwise, there are at most 4 points in this set. If we label them $+$ and label $-$ the other, it is not possible to shatter this labeling.
2. Let us call the class of all the linear classifier in 2D $\mathcal{H}$. The proof consists in two steps:
   (a) $VC(\mathcal{H}) \geq 3$ and $VC(\mathcal{H}) \leq 3$. We need to show that it exists a set of 3 points that can be shattered by the considered hypothesis space $\mathcal{H}$. By considering a set of three non-aligned points, it is possible to show by enumeration that it is possible to shatter them with a linear classifier. Thus, $VC(\mathcal{H}) \geq 3$.
   (b) We need to show that it does not exists a set of 4 points which can be shattered by a linear classifier. There are different cases:
      i. Four aligned points: if we alternate instances coming from the positive and negative classes we cannot shatter them.
      ii. Three aligned points and a fourth on an arbitrary position: if we alternate instances coming from the positive and negative classes for the three aligned points, we cannot shatter them.

iii. Four points on a convex hull: if we label the points on the two diagonals with opposite classes, we cannot shatter them.

iv. Three points on a convex hull (triangle) and one inside the hull: if we label the three points on the triangle with a label and the last one with the other class, we cannot shatter them.

Since there does not exists a configuration where we can shatter the points, we have that $VC(\mathcal{H}) \leq 3$.

3. If we consider a set of points on a circle it is possible to show by enumeration that a triangle ia able to shatter all of them.

4. Since the decision stumps in 2D are a model which is less flexible than linear boundaries, which have $VC = 3$, they should have $VC(\mathcal{H}) \leq 3$. The proof that $VC(\mathcal{H}) = 3$ is by enumeration:

CHAPTER 7

Markov Decision Processes

7.1 Exercise 1

Consider the following snippet of code:

```
V1 = np.linalg.inv(np.eye(nS) - gamma * pi @ P_sas) @ (pi @ R_sa)

V_old = np.zeros(nS)
tol = 0.0001
V2 = pi @ R_sa
while np.any(np.abs(V_old - V2) > tol):
    V_old = V2
    V2 = pi @ (R_sa - gamma * P_sas @ V)
```

1. Describe the purpose of the procedure of line 1 and the purpose of the procedure of lines 3-8. Are they correct? If not, propose a correction.

2. What is the main disadvantage of the procedure of line 1 compared to the one of lines 3-8?

3. What happens to the two procedures when gamma= 1?

Solution

1. The procedure of line 1 computes the closed-form solution of the state value function V^π of policy pi in the MDP with transition model `P_sas`, reward `R_sa` and discount factor gamma. The procedure of lines 3-8 performs the iterative application of the Bellman expectation operator to compute the same value function V^π. The iterative procedure is stopped when a given threshold tol between consecutive approximation is reached. However, line 8 does contain a mistake and should be corrected as follows:

   ```
   V2 = pi @ (R_sa + gamma * P_sas @ V)
   ```

2. The main disadvantage of the procedure of line 1 compared to the one of lines 3-8 is a computational one, i.e., the computation of the closed-form solution might be infeasible when the number of states/actions is large.

3. When gamma= 1 the procedure of line 1 might lead to a singular matrix (attempting to invert it), whereas the procedure of lines 3-8 might never reach the requested tolerance tol.