Machine Learning 1 — Final Exam

Preliminaries

- How to hand in:
  - write your answers on these exam sheets only;
  - write your immat **but not your name** on *every* page that you hand in.
  - hand in all exam sheets.
- The exam is open book. You may use all the material you want, while obeying the following rules:
  - you are not allowed to consult or communicate with other people, be it in the room or anywhere outside, except for with the examinators;
  - you must always place the screens of your computers and other used digital devices so that the examiners can see what you are doing;
  - failure to comply with these simple rules may lead to 0 points.

  In short, we will be as fair as we can, but expect the same from you in return.
- The exam is limited to 3 × 60 minutes.
- This exam consists of 14 pages, 7 sections, 20 problems. You can earn up to 72 points.

Problem 1 [2 points]. Fill in your immatriculation number on every sheet you hand in. As you will also write on these sheets, write it on this one, too. Make sure it is easily readable. Make sure you do **not** write your name on *any* sheet you hand in.

1 Linear (?) Classification

You are given a one-dimensional data set consisting of six points belonging to two classes. The set for the first class is $\{-1, 0, 1\}$ and for the second class $\{-.1, 0, .1\}$.

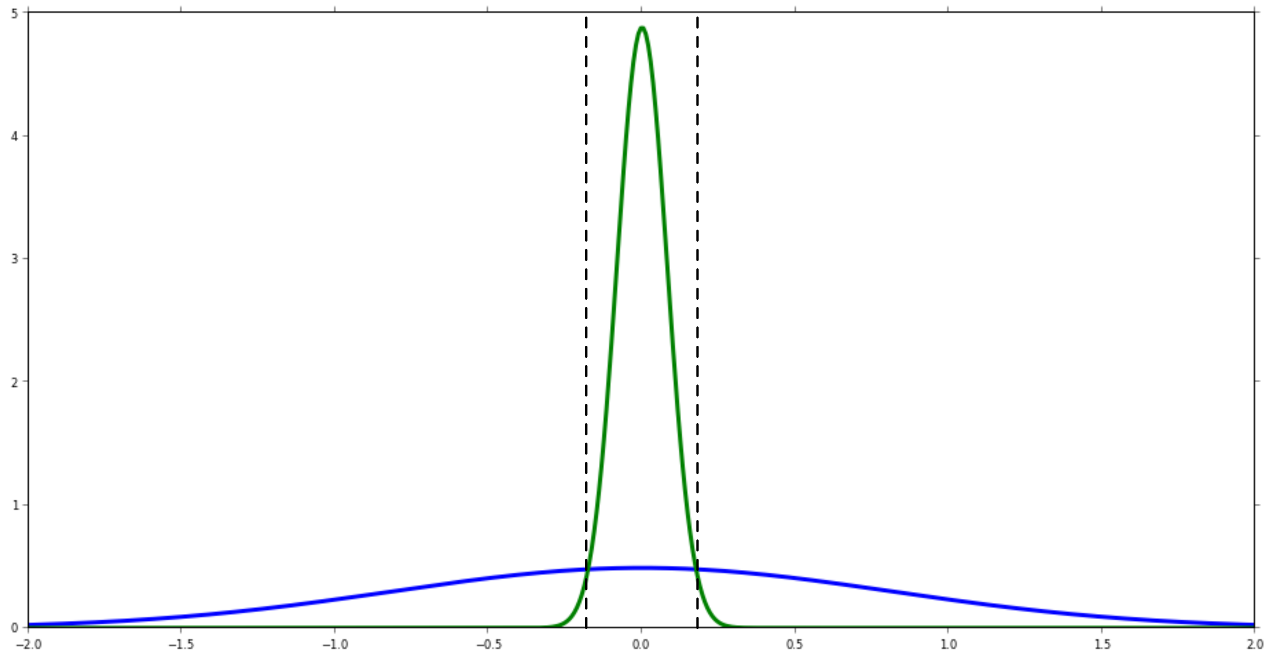
Problem 2 [2 points]. Consider a generative model where you want to model each class with a univariate Gaussian:

$$\begin{aligned} p(x|c_1) &= \mathcal{N}(\mu_1, \sigma_1^2) \\ p(x|c_2) &= \mathcal{N}(\mu_2, \sigma_2^2) \end{aligned}$$

Calculate the sufficient statistics μ_1, σ_1, μ_2 and σ_2 from the data set.

$$\begin{aligned} p(x|c_1) &= \mathcal{N}(0, 0.8165^2) \\ p(x|c_2) &= \mathcal{N}(0, 0.08165^2) \end{aligned}$$

Problem 3 [2 points]. Draw the points, the conditional densities and the decision boundaries **qualitatively** into a single diagram.



Problem 4 [2 points]. You are given the data set in the plot in Figure 1. Plot the decision boundaries of the following three models **qualitatively** into the plot:

- A generative model with Gaussian class conditionals where each class has its own mean and spherical covariance matrix,
- A logistic regression model,
- A linear model trained with soft zero-one loss.

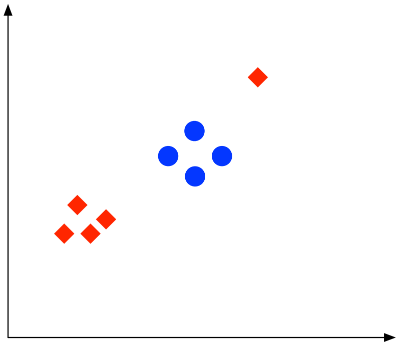
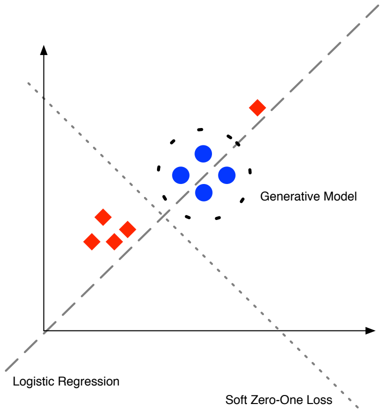


Figure 1: Some data points waiting to meet their decision boundaries.



2 Dice

Problem 5 [10 points]. As an avid player of board games you have a nice collection of non-standard dice: You have a 3-sided, 5-sided, 7-sided, 11-sided and 20-sided die. The 5 dice are treasured in a beautiful purple velvet bag. Without looking, a friend of yours randomly chooses a die from the bag and rolls a 6. What is the probability that the 11-sided die was chosen? What is the probability that the 20-sided die was used for the role? Show your work!

Now your friend rolls (with the same die!) an 18. What is the probability now that the die is 11-sided? What is the probability that it is 20-sided? Show your work!

Rolling a 6 eliminates the 3 and 5 sided dice (i.e. probability 0 for both dice). For the other 3 dice, the *likelihood* of rolling a 6 is 1/(number of sides). Let N denote the number of sides of a die. Using Bayes' Theorem, we can compute ($p(6)$ denotes the probability that a 6 is rolled):

$$p(N = n|6) = \frac{p(6|N = n) * 1/5}{p(6)} = \frac{1/(6n)}{p(6)}$$

For $n = 7, 11, 20$ the nominator is $0.028571, 0.018181, 0.01$, therefore $p(6) = 0.056753$ (the sum of these three numbers). This gives a probability of 0.32036 for $n = 11$ and 0.17620 for $n = 20$.

Rolling a 18 rules out all dice except the 20-sided one. Thus the probability for the 11 sided die is 0, for the 20 sided die it is 1.

3 Linear Regression

In Figure 2 you see 6 i.i.d noisy samples from an unknown function $f : \mathbb{R} \to \mathbb{R}$. Every sample (a black dot in the figure) is a pair $(x_i, z_i = f(x_i) + \varepsilon)$, where ε is some Gaussian noise, with mean zero and variance σ^2. We want to model this unknown function using polynomials of degree 5, that is $\hat{z}(x) = \sum_{i=0}^{5} w_i x^i$.

Because we have the 6 samples, we decide to use maximum likelihood estimation in order to find the parameters $\boldsymbol{w} = (w_0, w_1, w_2, w_3, w_4, w_5)$. We also know that MLE is prone to overfitting and therefore decide to use $\ell 2$ regularisation. Mathematically this is formulated as follows:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \sum_{j=1}^{6} \left[z_j - \sum_{i=0}^{5} w_i (x_j)^i \right]^2 + \lambda \sum_{i=1}^{5} w_i^2$$

where λ is the regularisation parameter. Note that w_0 is *not* regularised.

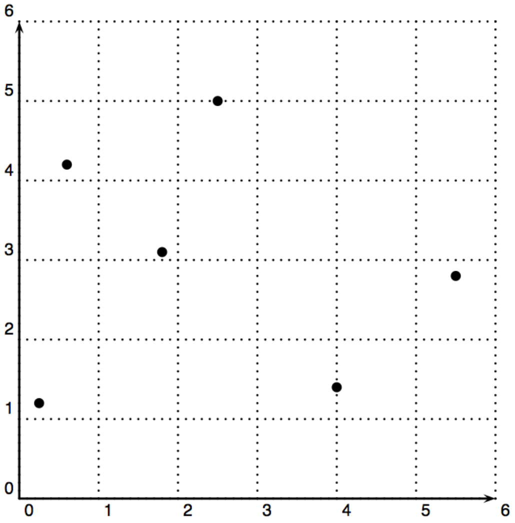


Figure 2: Figure for problems in section 3. 6 i.i.d samples from an unknown function f. The horizontal axis is the x-axis

Problem 6 [3 points]. Consider the 6 samples in Figure 2 again and draw (qualitatively) the solution with $\lambda = 0$ into Figure 2.

No regularization, thus the polynomial interpolates the 6 samples exactly.

Problem 7 [3 points]. What is the solution as $\lambda \to \infty$? Again, draw (qualitatively) your solution into Figure 2.

Only the bias is free, a horizontal line roughly in the middle of the six points.

Problem 8 [3 points]. Now, we also regularise w_0, i.e., the above optimisation problem is slightly changed to:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \sum_{j=1}^{6} \left[z_j - \sum_{i=0}^{5} w_i (x_j)^i \right]^2 + \lambda \sum_{i=0}^{5} w_i^2$$

What happens as $\lambda \to \infty$? Draw the *exact* solution again into Figure 2.

The x axis.

4 K-Means

Suppose we have some data points (the black dots) as in Figure 3. We want to apply K-Means to this data, using $k = 2$ and centres initialised to the two circled data points.

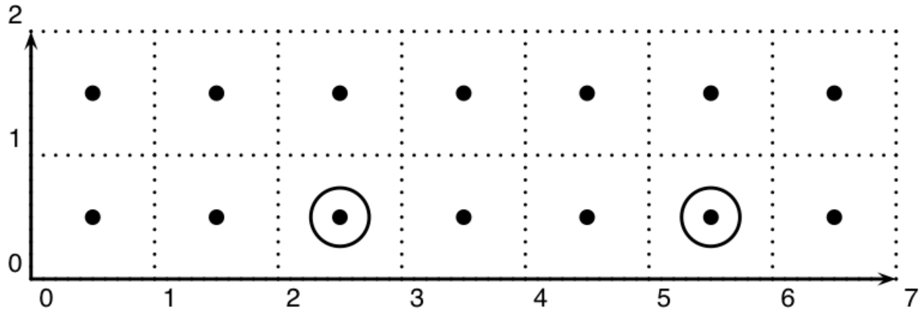


Figure 3: Cluster the black dots with K-Means. The two circles denote the initial guess for the two cluster centres, respectively.

Problem 9 [3 points]. What are the clusters after K-Means converges? Draw your solution into Figure 3, i.e., show the rough location of the centres (using a star-like shape) and show the clusters by drawing two bounding boxes around the points assigned to each centre.

The split into two clusters is along the vertical axis $x = 4$. Cluster center 1 is at (2,1), cluster center 2 is at (5.5, 1).

Problem 10 [3 points]. Give a *different* initialisation that will produce the same result. Draw your choice also into Figure 3, using triangles.

Lots of choices, e.g. choose the final centers. Or choose (3.5, 0.5) and (4.5, 0.5).

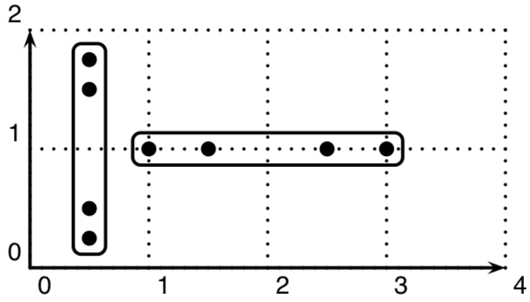


Figure 4: Can this clustering be produced by K-Means?

Problem 11 [4 points]. Dr. Iso Map claims, the clustering in Figure 4 (indicated by the bounding boxes) *results* from a K-Means run. Is this possible? Give reason(s) for your answer.

No, it can't be the result of a K-Means run. Consider the point at (1,1). It is assigned to the cluster with center (2,1), but is much closer to the cluster center at (0.5, 1). Thus the K-Means algorithm would not have converged yet.

5 Neural Networks

Zila could not pass on this offer… as engineer she always wanted to have a robot, and this was too good to be true.

On arrival of the package, Zila is irritated to only find the 3-joint arm itself—no manual, no control board, nothing. Fortunately, she recognises the connector of the robot: a standard ******[1] connector. So off she goes and purchases the corresponding computer interface, and can quickly connect to the robot.

The connector, she knows, allows for several interfaces. One is a simple position control interface: if one sends a command (q_1, q_2, q_3)—the angles of the three rotary joints—the robot moves to that position. But how can she use that to move the robot's hand to a position (x, y, z)? Zila does not know the kinematics of the robot.

How can she solve the inverse kinematics, i.e., find the function $f(x, y, z) = (q_1, q_2, q_3)$? Zila smartly decides to use a neural network to solve the problem. She randomly generates about $N = 1000$ joint positions (q^i_1, q^i_2, q^i_3) and records the corresponding points (x^i, y^i, z^i) where the hand ends. To represent the data, Zila takes a neural network

$$\mathcal{F}(x_1, x_2, x_3) = \left(\sum_{i=1}^{n} w_{1i}\sigma \left(\sum_{j=0}^{3} v_{ij}x_j \right), \sum_{i=1}^{n} w_{2i}\sigma \left(\sum_{j=0}^{3} v_{ij}x_j \right), \sum_{i=1}^{n} w_{3i}\sigma \left(\sum_{j=0}^{3} v_{ij}x_j \right) \right)$$

where $x_0 = 1$ and $\sigma(x) = \tanh(x)$. To determine the parameters v and w she minimises the error

$$E = \frac{1}{2} \sum_{i=1}^{L \le N} \left(f(x^i, y^i, z^i) - \mathcal{F}(x^i, y^i, z^i) \right)^2$$

using back-propagation.

Problem 12 [4 points]. What would happen if we would take $x_0 = 0.5$? What would happen if we would take $x_0 = 0$?

The result would be the same if $x_0 = 0.5$, but the corresponding parameters will be different; however, if $x_0 = 0$ a solution would, in general, not be found. It is the bias!

Problem 13 [4 points]. Explain why Zila's (standard) choice of error function means, that she implicitly assumes that her measurement error has a Gaussian distribution. Hint: consider what the corresponding likelihood function is.

The answer is given in the linear-regression slides: when the likelihood function is

$$p(\boldsymbol{z}|\boldsymbol{X}, \boldsymbol{w}, \sigma^2) = \prod^{N} \mathcal{N}(z_n|\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_n), \sigma^2)$$

then the non-constant term of the log likelihood is Zila's error function.

[1] We don't want to be commercial here.

Problem 14 [4 points]. At what value of E should Zila stop training?

She should divide the samples in training and test samples, and stop training at the lowest cross validation error.

Problem 15 [6 points]. It works! Zila finds an acceptable number n of hidden units with which she can get a reasonable representation of f. However, she finds that she *does* need many, many samples N to find her optimal value of E.

She has a great idea: rather than using $\sigma(x) = \tanh(x)$, she goes ahead and trains with $\sigma(x) = \sin(x)$. Why is this a good idea for learning the inverse kinematics of a robot with rotary joints?

Because the function described by rotary joints is a convolution of sin/cos functions. Exact approximation of a trig function can only be done with a trig function.

6 Stacking feature maps

Suppose you have found a feature map $\boldsymbol{\theta} : \mathbb{R}^n \to \mathbb{R}^m$ that transforms your data into a feature space in which a SVM with a Gaussian kernel works well. However computing the feature map $\boldsymbol{\theta}(\boldsymbol{x})$ is computationally expensive and luckily you discover an efficient method to compute the scalar product $K(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\theta}(\boldsymbol{x})^T \boldsymbol{\theta}(\boldsymbol{y})$ in your feature space without having to compute $\boldsymbol{\theta}(\boldsymbol{x})$ and $\boldsymbol{\theta}(\boldsymbol{y})$ explicitly.

Problem 16 [6 points]. Show how you can use the scalar product $K(\boldsymbol{x}, \boldsymbol{y})$ to efficiently compute the Gaussian kernel in your feature space, that is

$$K_g(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{\theta}(\boldsymbol{y}))$$

where

$$K_g(\boldsymbol{a}, \boldsymbol{b}) = \exp\left(- \frac{|\boldsymbol{a} - \boldsymbol{b}|^2}{2\sigma^2} \right)$$

is the Gaussian kernel.

By expanding the quadratic term and applying the definition of the $K(\boldsymbol{x}, \boldsymbol{y})$ we get

$$\begin{aligned} K_g(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{\theta}(\boldsymbol{y})) &= \exp\left(- \frac{|\boldsymbol{\theta}(\boldsymbol{x}) - \boldsymbol{\theta}(\boldsymbol{y})|^2}{2\sigma^2} \right) \\ &= \exp\left(- \frac{(\boldsymbol{\theta}(\boldsymbol{x}) - \boldsymbol{\theta}(\boldsymbol{y}))^T (\boldsymbol{\theta}(\boldsymbol{x}) - \boldsymbol{\theta}(\boldsymbol{y}))}{2\sigma^2} \right) \\ &= \exp\left(- \frac{\boldsymbol{\theta}(\boldsymbol{x})^T \boldsymbol{\theta}(\boldsymbol{x}) - 2\boldsymbol{\theta}(\boldsymbol{x})^T \boldsymbol{\theta}(\boldsymbol{y}) + \boldsymbol{\theta}(\boldsymbol{y})^T \boldsymbol{\theta}(\boldsymbol{y})}{2\sigma^2} \right) \\ &= \exp\left(- \frac{K(\boldsymbol{x}, \boldsymbol{x}) - 2K(\boldsymbol{x}, \boldsymbol{y}) + K(\boldsymbol{y}, \boldsymbol{y})}{2\sigma^2} \right) . \end{aligned}$$

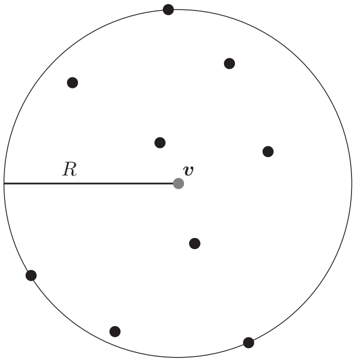
Thus we can compute $K_g(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{\theta}(\boldsymbol{y}))$ from $K(\boldsymbol{x}, \boldsymbol{y})$ using the above equation.

7 Constrained optimisation

Given l points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_l \in \mathbb{R}^N$ consider the problem of finding the ball, that is the set

$$B_R(\boldsymbol{v}) = \{\boldsymbol{u} : |\boldsymbol{u} - \boldsymbol{v}|^2 \leq R^2\} ,$$

with minimum radius R that contains all points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_l$. For example, in two dimensions this is the smallest circle that contains all given points as shown in the figure below.



Problem 17 [3 points]. Formulate the problem of finding the smallest ball $B_R(\boldsymbol{v})$ that contains all points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_l$ as a constrained optimisation problem. Note that *both* R and $\boldsymbol{v}$ are not given, but have to be found during the optimisation.

We minimize the squared radius R^2 and formulate the constraints using squared distances,

$$\begin{aligned} &\text{minimize } f_0(R, \boldsymbol{v}) = R^2 \\ &\text{subject to } f_i(R, \boldsymbol{v}) = (\boldsymbol{x_i} - \boldsymbol{v})^2 - R^2 \leq 0 \qquad i \in \{1, \ldots, l\} . \end{aligned}$$

Note that if we chose to minimize the objective function $f_0(R, \boldsymbol{v}) = R$ we need to add the constraint

$$f_{l+1}(R, \boldsymbol{v}) = -R \leq 0 .$$

This makes the following problems more difficult though.

Problem 18 [2 points]. Formulate the Lagrangian corresponding to this constrained optimisation problem.

The Lagrangian is given by

$$L(R, \boldsymbol{v}, \boldsymbol{\alpha}) = R^2 + \sum_{i=1}^{l} \alpha_i \left[(\boldsymbol{x_i} - \boldsymbol{v})^2 - R^2\right] .$$

Problem 19 [4 points]. Calculate the Lagange dual function and formulate the Lagrange dual problem. Do *not* attempt to solve the Lagrange dual problem.

We calculate the gradient of $L(R, \boldsymbol{v}, \boldsymbol{\alpha})$ w.r.t. $\boldsymbol{v}$ and set it to zero.

$$\begin{aligned} \nabla_{\boldsymbol{v}} L = -\sum_{i=1}^{l} 2\alpha_i(\boldsymbol{x_i} - \boldsymbol{v}) = 0 \\ \sum_{i=1}^{l} \alpha_i \boldsymbol{x_i} = \sum_{i=1}^{l} \alpha_i \boldsymbol{v} \\ \boldsymbol{v} = \frac{\sum_{i=1}^{l} \alpha_i \boldsymbol{x_i}}{\sum_{j=1}^{l} \alpha_j} \end{aligned}$$

We calculate the derivate of $L(R, \boldsymbol{v}, \boldsymbol{\alpha})$ w.r.t. R and set it to zero. This yields

$$\begin{aligned} \frac{\partial L}{\partial R} = 2R - 2\sum_{i=1}^{l} \alpha_i R = 0 \\ 1 - \sum_{i=1}^{l} \alpha_i = 0 \\ \sum_{i=1}^{l} \alpha_i = 1\,. \end{aligned}$$

Here we assumed that the problem is not degenerate, so at least two points are in different positions and thus $R \neq 0$. Substituting this into $\boldsymbol{v}$ yields

$$\boldsymbol{v} = \sum_{i=1}^{l} \alpha_i \boldsymbol{x_i}\,.$$

To calculate the Lagrange dual function $g(\boldsymbol{\alpha})$ we substitute $\boldsymbol{v}$ into $L(R, \boldsymbol{v}, \boldsymbol{\alpha})$ and apply $\sum \alpha_i = 1$. We get

$$g(\boldsymbol{\alpha}) = \sum_{i=1}^{l} \alpha_i \left(\boldsymbol{x_i} - \sum_{j=1}^{l} \alpha_j \boldsymbol{x_j} \right)^2 .$$

The Lagrange dual problem is given by

$$\begin{aligned} \text{maximize } g(\boldsymbol{\alpha}) = \sum_{i=1}^{l} \alpha_i \left(\boldsymbol{x_i} - \sum_{j=1}^{l} \alpha_j \boldsymbol{x_j} \right)^2 \\ \text{subject to } \alpha_i \geq 0 \qquad i \in \{1, \ldots, l\} \\ \sum_{i=1}^{l} \alpha_i = 1 \end{aligned}$$

Problem 20 [2 points]. In the two dimensional example shown in the figure above mark all points $\boldsymbol{x_i}$ which have their corresponding Lagrange multiplier $\alpha_i = 0$. Shortly explain how you chose these points.

All points not intersecting with the circle have $\alpha_i = 0$, i.e. their constraints are inactive. Their constraints are inactive because moving them by a small amount would not change the ball with the minimum radius.

More formally, the complementary slackness KKT condition

$$\alpha_i f_i(\boldsymbol{x_i}) = 0$$

yields

$$\alpha_i \left[(\boldsymbol{x_i} - \boldsymbol{v})^2 - R^2 \right] = 0\,.$$

For points $\boldsymbol{x_i}$ inside the circle we obviously have

$$(\boldsymbol{x_i} - \boldsymbol{v})^2 < R^2$$

and thus their corresponding Lagrange multiplier α_i must be zero.