True / False

a) False, Naive Bayes assumes that all features are conditionally independent of the class. It's this assumption that limits its performance. It doesn't have to do with knowing the distribution.

b) True Linear regression is convex. But to have 1 min that is the global min, it must be strictly convex. This occurs when H is positive semidefinite

$$\frac{\partial}{\partial \beta}\left(-2(X^T y) + 2X^T X \beta\right)$$

$H = 2X^T X > 0$ only if X is full rank
positive definite

1) convex → multiple optima, same value

2) strictly convex → 1 optima, that is global

c) False, counterexample is objective for ℓ_1-regularized linear regression. ℓ_1 norm is not differentiable but objective is still convex. sq err + l1 norm convex, sum of convex functions is convex.

d) True, sq'd error + ℓ_2 norm are both convex

sum of convex functions is convex

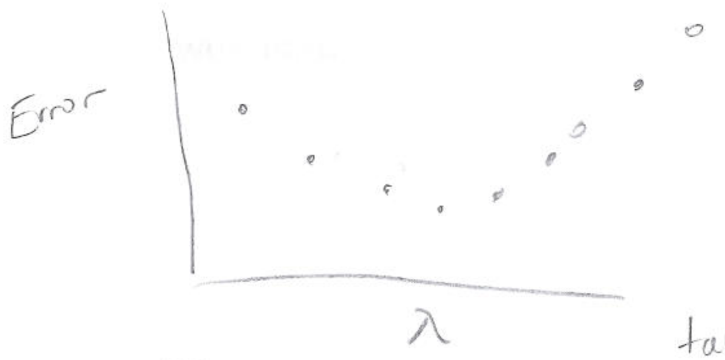
2. a) X^{train} → X^1 random 90% random 10% validation → error

→ X^2 → error

→ X^3 → error

⋮

X^{10}



take best lambda.

---

get test error from test set

b) In the dual formation of the SVM, features appear only as dot products, which are represented compactly by kernels

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

c) $\min \; x^2 + 1$

s.t. $(x-2)(x-4) \le 0$

$$L(x,\lambda) = x^2 + 1 + \lambda\left((x-2)(x-4)\right) \quad \text{where } \lambda \ge 0$$

$$= x^2 + 1 + \lambda(x^2 - 6x + 8)$$

$$= x^2 + 1 + \lambda x^2 - 6\lambda x + 8\lambda$$

$$= (\lambda+1)x^2 - 6\lambda x + 8\lambda + 1$$

To get the dual we $\max_{\lambda} \left(\min_{x} L(x,\lambda) \right)$

$$\frac{\partial L(x,\lambda)}{\partial x} = 2(\lambda+1)x - 6\lambda$$

$$0 = 2(\lambda+1)x - 6\lambda$$

$$x = \frac{\cancel{6}3\lambda}{\cancel{2}(\lambda+1)} = \frac{3\lambda}{\lambda+1}$$

point that min L via x.

b) In the dual of the SVM, features appear only at dot products, which are represented compactly by kernels

$$\max_{\alpha} \sum \alpha_i - \frac{1}{2} \sum \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

c) $L(x, \lambda) = x^2 + 1 + \lambda((x-2)(x-4))$ where $\lambda \geq 0$

$$= x^2 + 1 + \lambda(x^2 - 6x + 8)$$

$$= x^2 + 1 + \lambda x^2 - 6\lambda x + 8$$

$$= x^2 + 1 + \lambda x^2 - 6\lambda x + 8\lambda$$

$$= (\lambda + 1)x^2 - 6\lambda x + 8\lambda + 1$$

$$\max_{\lambda} \left(\min_{x} L(x, \lambda) \right)$$

$$\frac{\partial L(x,\lambda)}{\partial x} = 2(\lambda+1)x - 6\lambda$$

$$0 = 2(\lambda+1)x - 6\lambda$$

$$x = \frac{\cancel{6}\,3}{\cancel{2}(\lambda+1)} = \frac{3\lambda}{\lambda+1}$$

$$L(\lambda) = (\lambda+1)\left(\frac{3\lambda}{\lambda+1}\right)^2 - 6\lambda\left(\frac{3\lambda}{\lambda+1}\right) + 8\lambda + 1$$

$$= \frac{9\lambda^2 \cancel{(\lambda+1)}}{(\lambda+1)^{\cancel{2}}} - \frac{18\lambda^2}{\lambda+1} + 8\lambda + 1$$

$$= \frac{9\lambda^2}{(\lambda+1)} - \frac{18\lambda^2}{\lambda+1} + 8\lambda + 1$$

$$= \frac{-9\lambda^2}{\lambda+1} + 8\lambda + 1$$

$$L(\lambda) = (\lambda+1)\cdot\left(\frac{3\lambda}{\lambda+1}\right)^2 - 6\lambda\left(\frac{3\lambda}{\lambda+1}\right) + 8\lambda + 1$$

$$= \frac{9\lambda^2 \cancel{(\lambda+1)}}{(\lambda+1)^{\cancel{2}}} - \frac{18\lambda^2}{\lambda+1} + 8\lambda + 1$$

$$= \frac{9\lambda^2}{(\lambda+1)} - \frac{18\lambda^2}{\lambda+1} + 8\lambda + 1$$

$$= \frac{-9\lambda^2}{\lambda+1} + 8\lambda + 1$$

$$\max \quad -\frac{9\lambda^2}{\lambda+1} + 8\lambda + 1$$

$$\text{s.t.} \quad \lambda \geq 0.$$

Naive Bayes

	$P(x_a=1 \mid y)$	$P(x_b=1 \mid y)$	x_c	x_d	x_e	x_f	x_g
$y=1$	0	0	1/2	1	0	0	1/2
-1	1	1/2	0	0	1/2	1	0

$P(y=1 \mid z1) = 0.$ $\qquad P(y=-1)$

$P(y=-1 \mid z1) = 1 \times \frac{1}{2} \times 1 \times \frac{1}{2} \times 1 \times 1 \times \frac{1}{2}$

$= \frac{1}{8}$

$y=-1$

$P(y=1 \mid z2) = 1 \times 0$

$P(y=-1 \mid z2) = 0$

$x_1 =$	0	0	0	1	0	0	1	$y=1$
$x_2 =$	0	0	1	1	0	0	0	$y=1$
$x_3 =$	1	1	0	0	0	1	0	$y=-1$
$x_4 =$	1	0	0	0	1	1	0	$y=-1$
$x_5 =$	1	1	1	1	1	1	1	$y=1$
$x_6 =$	0	0	0	0	0	0	0	$y=1$
$x_7 =$	1	1	1	1	1	1	1	$y=-1$
$x_8 =$	0	0	0	0	0	0	0	$y=-1$

	x_a	x_b	x_c	x_d	x_e	x_f	x_g
$y=1$	1/4	1/4	2/4	3/4	1/4	1/4	2/4
$y=-1$	3/4	2/4	1/4	1/4	2/4	3/4	1/4

Naive Bayes

$$P(y=1 \mid z1) = \left(\frac{1}{4} \cdot \frac{3}{4} \cdot \frac{2}{4} \cdot \frac{1}{4} \cdot \frac{3}{4} \cdot \frac{1}{4} \cdot \frac{2}{4}\right) \cdot \underbrace{\frac{1}{2}}_{P(y=1)} = \frac{36}{2^{15}}$$

$$P(y=-1 \mid z1) = \left(\frac{3}{4} \cdot \frac{2}{4} \cdot \frac{3}{4} \cdot \frac{3}{4} \cdot \frac{2}{4} \cdot \frac{3}{4} \cdot \frac{3}{4}\right) \cdot \frac{1}{2} = \frac{3^5 \cdot 2^2}{2^{15}}$$

$y = -1$

$$P(y=1 \mid z2) = \left(\frac{3}{4} \cdot \frac{1}{4} \cdot \frac{2}{4} \cdot \frac{1}{4} \cdot \frac{3}{4} \cdot \frac{1}{4} \cdot \frac{2}{4}\right)\frac{1}{2} = \frac{36}{2^{15}}$$

$$P(y=1 \mid z2) = \left(\frac{1}{4} \cdot \frac{2}{4} \cdot \frac{1}{4} \cdot \frac{3}{4} \cdot \frac{2}{4} \cdot \frac{3}{4} \cdot \frac{1}{4}\right)\frac{1}{2} = \frac{36}{2^{15}}$$

tie

Perceptron

Start

$w = (0,0,0,0,0,0,0)$

$y_1 [\langle w, x_1 \rangle] = 0 \leq 0$? Yes

$w = w + y_1 x_1 = (0,0,0,1,0,0,1)$

$y_2 [\langle w, x_2 \rangle] = 0 \leq 0$? Yes

$w = w + y_2 x_2$

$= (0,0,0,1,0,0,1) + (-1,-1,0,0,0,-1,0)$

$= (-1,-1,0,1,0,-1,1)$

$y_3 [\langle w, x_3 \rangle] = 1 \leq 0$? No

no update

$y_4 [\langle w, x_4 \rangle] = (-1-1)-1 = 2 \leq 0$? No

No update

$y_5 \langle w, x_5 \rangle = (-1+-1)-1 = 2 \leq 0$? No

No update

$w = (-1,-1,0,1,0,-1,1)$

SVMs

1. Write down the problem

$$\min_{w,b} \ \frac{1}{2}\|w\|^2 + C\sum \xi_i$$

$$\text{s.t.} \quad y_i[\langle w, x_i\rangle + b] \geq 1 - \xi_i$$

$$\xi_i \geq 0$$

for large values of C, penalizing shrinking the margin heavily.
that is penalizing misclassifyed points
∴ decision boundary will seperate data perfectly if possible.

2. C=0, not penalizing misclassified points at all
penalty is low, so we can misclassify a few while maximizing the margin btw most of the points

3. Warning was don't trust any specific data point too much, so we prefer $C \approx 0$

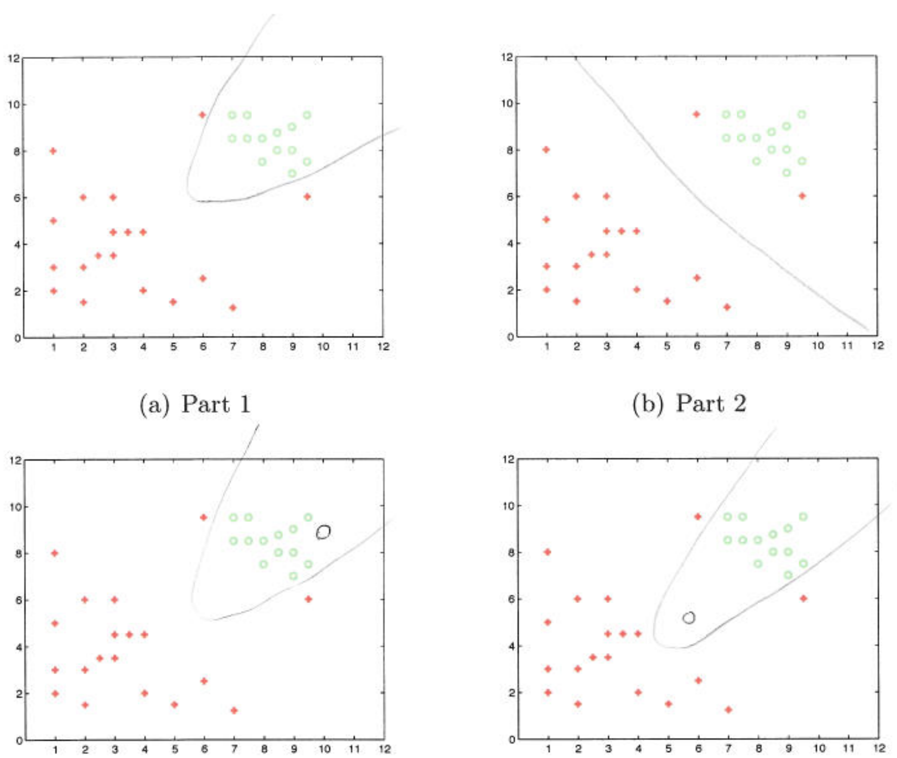
4. Correctly classified by the original classifier, will not be a support vector

5. C is larg, adding a point that is incorrectly classified by the original boundary would force the boundary to move.

3. [2 points] Which of the two cases above would you expect to work better in the classification task? Why?

4. [3 points] Draw a data point which will not change the decision boundary learned for very large values of C. Justify your answer.

5. [3 points] Draw a data point which will significantly change the decision boundary learned for very large values of C. Justify your answer.



(a) Part 1 (b) Part 2

(c) Part 4 (d) Part 5

Figure 2: Draw your solutions for Problem 2 here.

Conditional Independence, MLE/MAP, Prob.

1. Use Chain Rule

$$P(A_n \ldots A_1) = P(A_n | A_{n-1} \ldots A_1) \, P(A_{n-1} \ldots A_1)$$

$$P(X, Y | Z) = \underbrace{P(X | Y, Z)} P(Y|Z)$$

$$= P(X|Z) \, P(Y|Z)$$

2.

$$\mathcal{L} = \prod_{i=1}^{n} \frac{\theta^{y_i} e^{-\theta}}{y_i!}$$

$$\log \mathcal{L} = \log \prod_{i=1}^{n} \frac{\theta^{y_i} e^{-\theta}}{y_i!}$$

$$= \sum_{i=1}^{n} \left(y_i \log \theta - \theta - \log y_i! \right)$$

$$= \log \theta \sum_{i=1}^{n} y_i - n\theta - \sum_{i=1}^{n} \log y_i!$$

3. $P(\text{correct} | \text{answer}) = 1$ $\quad P(\text{answer}) = p$

$P(\text{correct} | \text{guess}) = \frac{1}{m}$ $\quad P(\text{guesses}) = 1 - p$

$$P(\text{answer} | \text{correct}) = \frac{P(\text{correct} | \text{answer}) \, P(\text{answer})}{P(\text{correct})}$$

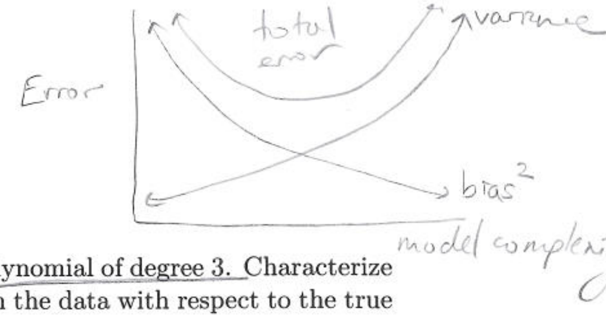
$$= \frac{P(\text{correct} | \text{answer}) \, P(\text{answer})}{P(\text{correct} | \text{answer}) \, P(\text{answer}) + P(\text{correct} | \text{guess}) \, (P(\text{guess}))}$$

$$= \frac{1 \cdot p}{1 \cdot p + \frac{1}{m}(1-p)}$$

$$= \frac{p}{p + \frac{1}{m}(1-p)}$$

Err = $\text{Bias}^2 = \left(E[\hat{f}(x)] - f(x)\right)^2$

$\text{Var} = \left[\hat{f}(x) - E[\hat{f}(x)]\right]^2$



4 Bias-Variance Decomposition (12 pts)

1. (6 pts) Suppose you have regression data generated by a polynomial of degree 3. Characterize the bias-variance of the estimates of the following models on the data with respect to the true model by circling the appropriate entry.

SKIP

	Bias	Variance
Linear regression	low/high	low/high
Polynomial regression with degree 3	low/high	low/high
Polynomial regression with degree 10	low/high	low/high

2. Let $Y = f(X) + \epsilon$, where ϵ has mean zero and variance σ_ϵ^2. In k-nearest neighbor (kNN) regression, the prediction of Y at point x_0 is given by the average of the values Y at the k neighbors closest to x_0.

   (a) (2 pts) Denote the ℓ-nearest neighbor to x_0 by $x_{(\ell)}$ and its corresponding Y value by $y_{(\ell)}$. Write the prediction $\hat{f}(x_0)$ of the kNN regression for x_0 in terms of $y_{(\ell)}, 1 \le \ell \le k$.

   $$\hat{f}(x_0) = \frac{1}{K}\sum_{\ell=1}^{K} y_{(\ell)}$$

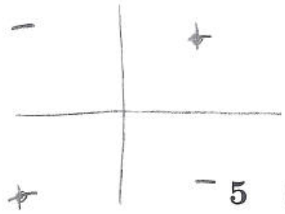
   (b) (2 pts) What is the behavior of the bias as k increases?

   decreases

   solutions online are wrong

   (c) (2 pts) What is the behavior of the variance as k increases?

   increases



5 Support Vector Machine (12 pts)

Consider a supervised learning problem in which the training examples are points in 2-dimensional space. The positive examples are $(1,1)$ and $(-1,-1)$. The negative examples are $(1,-1)$ and $(-1,1)$.

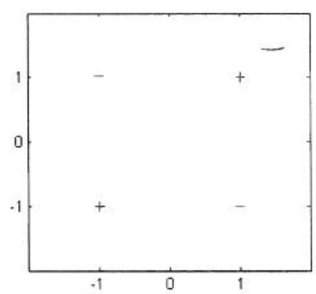
1. (1 pts) Are the positive examples linearly separable from the negative examples in the original space?

   No

2. (4 pts) Consider the feature transformation $\phi(x) = [1, x_1, x_2, x_1x_2]$, where x_1 and x_2 are, respectively, the first and second coordinates of a generic example x. The prediction function is $y(x) = w^T * \phi(x)$ in this feature space. Give the coefficients, w, of a maximum-margin decision surface separating the positive examples from the negative examples. (You should be able to do this by inspection, without any significant computation.)

   $w = (0,0,0,1)^T$

3. (3 pts) Add one training example to the graph so the total five examples can no longer be linearly separated in the feature space $\phi(x)$ defined in problem 5.2.



4. (4 pts) What kernel $K(x, x')$ does this feature transformation ϕ correspond to?

   $[1, x_1, x_2, x_1 x_2]$ $\quad \phi(x)\cdot\phi(x')$

   $[1, x_1', x_2', x_1' x_2']$

$1 + x_1x_1' + x_2x_2' + x_1x_1'x_2x_2'$

6 Generative vs. Discriminative Classifier (20 pts)

Consider the binary classification problem where class label $Y \in \{0, 1\}$ and each training example X has 2 binary attributes $X_1, X_2 \in \{0, 1\}$.

In this problem, we will always assume X_1 and X_2 are conditional independent given Y, that the class priors are $P(Y = 0) = P(Y = 1) = 0.5$, and that the conditional probabilities are as follows:

$P(X_1\|Y)$	$X_1 = 0$	$X_1 = 1$
$Y = 0$	0.7	0.3
$Y = 1$	0.2	0.8

$P(X_2\|Y)$	$X_2 = 0$	$X_2 = 1$
$Y = 0$	0.9	0.1
$Y = 1$	0.5	0.5

The expected error rate is the probability that a classifier provides an incorrect prediction for an observation: if Y is the true label, let $\hat{Y}(X_1, X_2)$ be the predicted class label, then the expected error rate is

$$P_{\mathcal{D}}\left(Y = 1 - \hat{Y}(X_1, X_2)\right) = \sum_{X_1=0}^{1} \sum_{X_2=0}^{1} P_{\mathcal{D}}\left(X_1, X_2, Y = 1 - \hat{Y}(X_1, X_2)\right).$$

Note that we use the subscript $\mathcal{D}$ to emphasize that the probabilities are computed under the true distribution of the data.

*You don't need to show all the derivation for your answers in this problem.

1. (4 pts) Write down the naïve Bayes prediction for all the 4 possible configurations of X_1, X_2. The following table would help you to complete this problem.

X_1	X_2	$P(X_1, X_2, Y = 0)$	$P(X_1, X_2, Y = 1)$	$\hat{Y}(X_1, X_2)$
0	0	0.7 · 0.9 · 0.5	0.2 · 0.5 · 0.5	0
0	1	0.7 · 0.1 · 0.5	0.2 · 0.5 · 0.5	1
1	0	0.3 · 0.9 × 0.5	0.8 · 0.5 · 0.5	1
1	1	0.3 · 0.1 × 0.5	0.8 · 0.5 · 0.5	1

2. (4 pts) Compute the expected error rate of this naïve Bayes classifier which predicts Y given both of the attributes $\{X_1, X_2\}$. Assume that the classifier is learned with infinite training data.

skip

(a) By regularizing w_2 **[3 pts]** Increases

By regularizing w_2, the boundary can rely less and less on x_2 and thus boundary becomes more vertical.

(b) By regularizing w_1 **[3 pts]** Same.

By regularizing w_1, the boundary can rely less and less on x_1, and thus boundary becomes more horizontal. That's ok b/c training data can be separated by horizontal linear separator

(c) By regularizing w_0 **[3 pts]**

Increase, When we regularize w_0, then the boundary will eventually go through the origin

Best we can get is one error

2. If we change the form of regularization to L1-norm (absolute value) and regularize w_1 and w_2 only (but not w_0), we get the following penalized log-likelihood

$$\max \sum_{i=1}^{n} \log P(y_i|x_i, w_0, w_1, w_2) - C(|w_1| + |w_2|).$$

Consider again the problem in Figure 1 and the same linear logistic regression model $P(y = 1|\vec{x}, \vec{w}) = g(w_0 + w_1x_1 + w_2x_2)$.

(a) **[3 pts]** As we increase the regularization parameter C which of the following scenarios do you expect to observe? (Choose only one) Briefly explain your choice:

( ) First w_1 will become 0, then w_2.

( ) First w_2 will become 0, then w_1.

( ) w_1 and w_2 will become zero simultaneously.

( ) None of the weights will become exactly zero, only smaller as C increases.

1) we can classify with zero error on x_2 alone so w_1 goes to zero. Note absolute value reg. ensures it goes exactly to zero

2) as C increases, we pay higher and higher cost for w_2 so it eventually goes to zero

Skip

(b) [**3 pts**] For very large C, with the same L1-norm regularization for w_1 and w_2 as above, which value(s) do you expect w_0 to take? Explain briefly. (Note that the number of points from each class is the same.) (You can give a range of values for w_0 if you deem necessary).

Skip

(c) [**3 pts**] Assume that we obtain more data points from the '+' class that corresponds to y=1 so that the class labels become unbalanced. Again for very large C, with the same L1-norm regularization for w_1 and w_2 as above, which value(s) do you expect w_0 to take? Explain briefly. (You can give a range of values for w_0 if you deem necessary).

Kernel Regression

1. $$\frac{\partial J(\beta)}{\partial \beta)} = -2A^T W (Y - A\beta)$$

$$\frac{\partial}{\partial s}(x - As)^T W (x - As) = -2A^T W (x - As)$$

$$0 = -2A^T W Y + 2A^T W A \beta$$

$$A^T W Y = A^T W A \beta$$

$$\hat{\beta} = (A^T W A)^{-1} A^T W Y$$

2. when $A^T W A$ is invertible, full rank

3. Gradient Descent

$$x^{(k+1)} = x^{(k)} - t_{k+1} \nabla f(x^{(k)})$$

$$\beta^{(t+1)} = \beta^{(t)} - \underset{\text{step size}}{\alpha} \frac{\partial J(\beta)}{\partial \beta}$$

$$= \beta^{(t)} - \alpha(-2A^T W (Y - A\beta))$$

$$= \beta^{(t)} - \alpha A^T W (A\beta - Y)$$

4. $$Y = \beta_1 + \beta_2 X + \epsilon$$

$$\epsilon \sim N(0, \sigma_i^2)$$

$$\sigma_i^2 \propto \frac{1}{W_i(x)}$$

5. 1 advantage → no strict assumptions on the form of the underlying distribution or regression function

1 disadvantage → computationally expensive
large # of training examples