HOMEWORK 5:
WRITTEN EXERCISE PART

>>XX<<
>>XXX<<

Part 1: Required Exercises

1 Conditional Independence [5 pts]

Consider three binary variables $a, b, c \in \{0, 1\}$ having the joint distribution given in Table 8.2.

a	b	c	$p(a,b,c)$
0	0	0	0.192
0	0	1	0.144
0	1	0	0.048
0	1	1	0.216
1	0	0	0.192
1	0	1	0.064
1	1	0	0.048
1	1	1	0.096

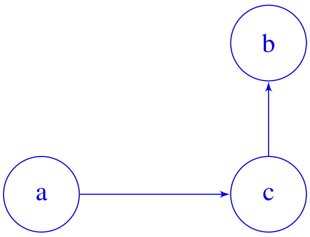
(a) Show by direct evaluation that this distribution has the property that a and b are marginally dependent, so that $p(a, b) \neq p(a)p(b)$, but that they become independent when conditioned on c, so that $p(a, b|c) = p(a|c)p(b|c)$ for both $c = 0$ and $c = 1$.
(b) Evaluate the distribution $p(a), p(b|c)$, and $p(c|a)$ corresponding to the joint distribution given in the table. Hence show by direct evaluation that $p(a, b, c) = p(a)p(c|a)p(b|c)$. Draw the corresponding Bayesian network.
(a) According to the table, we can get
$p(a = 0) = 0.192 + 0.144 + 0.048 + 0.216 = 0.6$
$p(a = 1) = 0.192 + 0.064 + 0.048 + 0.096 = 0.4$
$p(b = 0) = 0.192 + 0.144 + 0.192 + 0.064 = 0.592$
$p(b = 1) = 0.048 + 0.216 + 0.048 + 0.096 = 0.408$
$p(a = 0, b = 0) = 0.192 + 0.144 = 0.336$, but $p(a = 0)p(b = 0) = 0.3552$,
Therefore, we can see $p(a = 0, b = 0) \neq p(a = 0)p(b = 0)$, So a and b are marginally dependent.
But when conditioned on c, $p(c = 0) = 0.48$,
$p(a = 0, b = 0|c = 0) = 0.192 \div p(c = 0) = 0.4$, $p(a = 0|c = 0) = 0.5$, $p(b = 0|c = 0) = 0.8$
$p(a = 0, b = 1|c = 0) = 0.048 \div p(c = 0) = 0.1$, $p(a = 0|c = 0) = 0.5$, $p(b = 1|c = 0) = 0.2$
$p(a = 1, b = 0|c = 0) = 0.192 \div p(c = 0) = 0.4$, $p(a = 1|c = 0) = 0.5$, $p(b = 0|c = 0) = 0.8$
$p(a = 1, b = 1|c = 0) = 0.048 \div p(c = 0) = 0.1$, $p(a = 1|c = 0) = 0.5$, $p(b = 1|c = 0) = 0.2$
We can see that $p(a, b|c = 0) = p(a|c = 0)p(b|c = 0)$.
Similarly, for $p(c = 1) = 0.52$, and also satisfy $p(a, b|c = 1) = p(a|c = 1)p(b|c = 1)$, So they are marginally independent.

(b)

$p(a)$	t	f
	0.4	0.6

$p(c\|a)$	a	t	f
	1	0.4	0.6
	0	0.4	0.6

$p(b\|c)$	c	t	f
	1	0.6	0.4
	0	0.2	0.8



2 Information Gain [5 pts]

Consider the following training set with two boolean features and one continuous feature.

	A	B	C	Class
Instance 1	F	T	120	Benign
Instance 2	T	F	1090	Benign
Instance 3	T	T	245	Malignant
Instance 4	F	F	589	Malignant
Instance 5	T	T	877	Malignant

(a) How much information about the class is gained by knowing whether or not the value of feature C is less than 475?
(b) How much information about the class is gained by knowing whether or not the value of features A and B are different?
(a) Before division, classes are 2 Benign and 3 Malignant, $H(Y) = -\frac{2}{5}log_2\frac{2}{5} - \frac{3}{5}log_2\frac{3}{5} = 0.971$.
low($C < 475$) part, classes are 1 Benign and 1 Malignant, $H(Y|low) = -\frac{1}{2}log_2\frac{1}{2} - \frac{1}{2}log_2\frac{1}{2} = 1.0$;
high($C \geq 475$) part, classes are 1 Benign and 2 Malignant.$H(Y|high) = -\frac{1}{3}log_2\frac{1}{3} - \frac{2}{3}log_2\frac{2}{3} = 0.918$;
Therefore, the information gain is $H(Y) - (\frac{2}{5}H(Y|low) + \frac{3}{5}H(Y|high)) = 0.0202$

(b) for different($A \neq B$) part, classes are 2 Benigns.$H(Y|\text{different}) = 0$
for same($A = B$) part, classes are 3 Malignants.$H(Y|\text{same}) = 0$
Therefore, the information gain equals to $H(Y)$, information gain $= 0.971$

3 k-Nearest Neighbor [5 pts]

Suppose we want to learn a k-nearest neighbor model with the following data set and we are using Leave One Out Cross Validation (LOOCV) to select k. What would LOOCV pick: $k = 1$, or $k = 2$, or $k = 3$. Use Manhattan distance for calculations.

	Feature 1	Feature 2	Class
Instance 1	2	3	Positive
Instance 2	4	4	Positive
Instance 3	4	5	Negative
Instance 4	6	3	Positive
Instance 5	8	3	Negative
Instance 6	8	4	Negative

LOOCV would pick $k = 1$, since $k = 1$ acheives the best accuracy, and it is also stable and faster. $k = 1$: accuracy is 0.5

fold1(Instance 1: Positive): nearest is Instance 2(distance: 3), so predict Class of Instance 1 is Positive.
fold2(Instance 2: Positive): nearest is Instance 3(distance: 1), so predict Class of Instance 2 is Negative.
fold3(Instance 3: Negative): nearest is Instance 2(distance: 1), so predict Class of Instance 3 is Positive.
fold4(Instance 4: Positive): nearest is Instance 5(distance: 2), so predict Class of Instance 4 is Negative.
fold5(Instance 5: Negative): nearest is Instance 6(distance: 1), so predict Class of Instance 5 is Negative.
fold6(Instance 6: Negative): nearest is Instance 5(distance: 1), so predict Class of Instance 6 is Negative.

$k = 2$: accuracy is 0.5
p.s. Tie Breaking Rule is to choose instance with the smallest index, if the output meets a tie(exactly same number of negatives and positives), also choose the output of the instance with the smallest index.

fold1: nearest 2, 3/4. Break the tie, choose 2,3. smallest is 2. output is Positive.
fold2: nearest 3, 1/4. Break the tie, choose 3,1. smallest is 1. output is Positive.
fold3: nearest 2, 1/4. Break the tie, choose 2,1. output is Positive.
fold4: nearest 5, 6/2. Break the tie, choose 5,2. smallest is 2. output is Positive.
fold5: nearest 6, 4. smallest is 4. output is Positive.
fold6: nearest 5, 4. smallest is 4. output is Positive.

$k = 3$: accuracy is $\frac{1}{3}$
fold1: nearest 2,3,4, output is Positive.
fold2: nearest 3,1,4, output is Positive.
fold3: nearest 2,1,4, output is Positive.
fold4: nearest 5,6,2, output is Negative.
fold5: nearest 6,4,2, output is Positive.
fold6: nearest 5,4,2, output is Positive.

4 Nearest Neighbor Regression [5 points]

Given data points $x_1 = (-1, 0), x_2 = (0, 0), x_3 = (0, 1)$ in the 2-dimensional Euclidean space and their corresponding labels $y_1 = 1, y_2 = 2, y_3 = 3$, use weighted 2-Nearest Neighbor to compute the label for $x = (1, 1)$. Here the weighted 2-Nearest Neighbor estimate is

$$f(x) = \frac{\sum_{i=1}^{2} w_i y_{(i)}}{\sum_{i=1}^{2} w_i},$$

where the weight $w_i = 1/i$ and $y_{(i)}$ is the label of the i-th nearest neighbor.
The euclidean distance between x and x_1, x_2, x_3 are:
$d(x, x1) = \sqrt{5}$, $d(x, x2) = \sqrt{2}$, $d(x, x1) = 1$
So the nearest two neighbors are $y(1) = y_3 = 3$, $y(2) = y_2 = 2$; and $w_1 = 1$, $w_2 = 0.5$,
Then the output of x is
$f(x) = \frac{w_1 y(1) + w_2 y(2)}{w_1 + w_2} = \frac{8}{3} \approx 2.667$

5 Evaluation [5 points]

Consider the following confusion matrix of a 2-class problem.

	actual positive	actual negative
predict positive	60	30
predict negative	50	60

Table 1: Confusion matrix of a 2-class problem. There are 200 instances in total.

Compute the following: accuracy, error, precision, recall.
accuraccy = $\frac{TP+FN}{TP+FN+TN+FP} = \frac{60+60}{60+30+50+60} = 0.60$
error = $1-$ accuracy $= 0.40$

precision = $\frac{TP}{TP+FP} = \frac{60}{60+30} = 0.667$
recall = $\frac{TP}{TP+FN} = \frac{60}{60+50} = 0.545$

6 Logistic Regression [10 points]

Let $f(x) = \sigma(w^\top x)$ where $w = (1, 2)$ and σ is the sigmoid function $\sigma(z) = 1/(1 + \exp(-z))$. Compute the gradient ∇f at the point $x = (3, 4)$.

The loss function of Logistic Regression $f(x)$ is
$\hat{L}(w) = -\frac{1}{m}(\sum(ylog(\sigma(w^\top x)) + (1-y)log(1-\sigma(w^\top x))))$, here $m = 1$
Since the gradient of $\sigma(a)$ is $\sigma'(a) = \sigma(a)(1-\sigma(a))$
Gradient of $f(x)$ is $\nabla f = \frac{\partial \hat{L}(w)}{\partial w} = -y(1-\sigma(w^\top x)) + (1-y)\sigma(w^\top x)$
therefore, the gradient at the point $x = (3, 4)$ is
when y(x)=1, $\nabla f = -y(1-\sigma) = 1 - \sigma = -1.67 \times 10^{-5}$
when y(x)=0, $\nabla f = (1-y)\sigma = \sigma = 1.67 \times 10^{-5}$

7 Maximum A Posterior [10 points]

Given data points $\{x_i, 1 \le i \le n\}$ from the Gaussian distribution $N(\mu, I)$ where the mean μ is unknown. Use the prior $p(u) = N(x_0, I)$ and compute the Maximum A Posterior estimation of μ.

$\mu^{MAP} = \arg\max\limits_{\mu} \prod\limits_{i=1}^{n} p(x_i|\mu)p(\mu)$, where $p(\mu)$ is the prior distribution of μ.

After using log, it becomes $\mu^{MAP} = \arg\max\limits_{\mu}(\sum\limits_{1}^{n}(log(p(x_i|\mu))) + log(p(\mu)))$

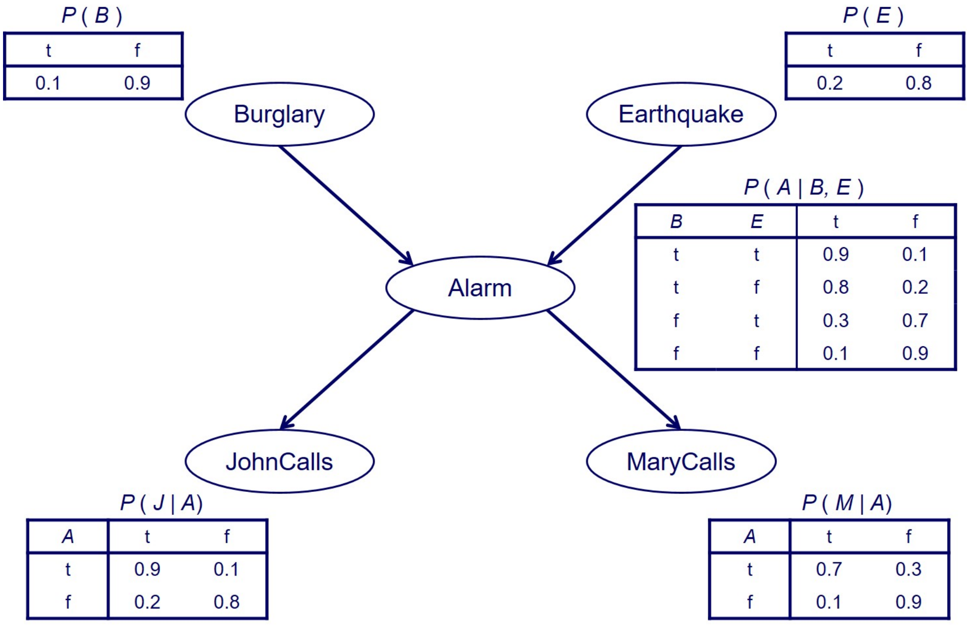
Consider the prior $p(u) = N(x_0, I)$, and $\{x_i, 1 \le i \le n\}$ come from the Gaussian distribution $N(\mu, I)$,

Let $\frac{\partial \mu^{MAP}}{\partial \mu} = \sum\limits_{i=1}^{n}(\frac{\mu - x_i}{I}) + \frac{\mu - \mu_0}{I} = 0$, so we can get

$\mu = \frac{\mu_0 + \sum\limits_{i=1}^{n} x_i}{n+1}$

8 Bayesian Networks [10 points]

Consider the Bayesian Network in Figure 1.



P (B)

t	f
0.1	0.9

P (E)

t	f
0.2	0.8

P (A | B, E)

B	E	t	f
t	t	0.9	0.1
t	f	0.8	0.2
f	t	0.3	0.7
f	f	0.1	0.9

P (J | A)

A	t	f
t	0.9	0.1
f	0.2	0.8

P (M | A)

A	t	f
t	0.7	0.3
f	0.1	0.9

Figure 1: A Bayesian Network example.

Compute $P(B=t,E=f,A=f,J=t,M=t)$ and $P(B=t,E=f,A=f,J=t|M=t)$.
$P(B=t,E=f,A=f,J=t,M=t)$
$=P(B=t)P(E=f)P(A=f|B=t,E=f)P(J=t|A=f)P(M=t|A=f)$
$=0.1\times 0.8\times 0.2\times 0.2\times 0.1=0.00032$

$P(B=t,E=f,A=f,J=t|M=t)=\frac{P(B=t,E=f,A=f,J=t,M=t)}{P(M=t|A=f)+P(M=t|A=t)}$
$P(M=t|A=t)=0.7\times(0.02\times 0.9+0.08\times 0.8+0.18\times 0.3+0.72\times 0.1)=0.0208$
$P(M=t|A=f)=0.1\times(0.02\times 0.1+0.08\times 0.2+0.18\times 0.7+0.72\times 0.9)=0.5544$
Therefore, $P(B=t,E=f,A=f,J=t|M=t)=0.000556$

9 Bayes Network: Sparse Candidate Algorithm [10 pts]

Suppose we wish to construct a Bayes Network for 3 features X, Y, and Z using Sparse Candidate algorithm. We are given data from 100 independent experiments where each feature is binary and takes value T or F. Below is a table summarizing the observations of the experiment:

X	Y	Z	Count
T	T	T	36
T	T	F	4
T	F	T	2
T	F	F	8
F	T	T	9
F	T	F	1
F	F	T	8
F	F	F	32

(a) Suppose we wish to compute a single candiate parent for Z. In the first round of the sparse Candidate algorithm, we compute the mutual information between Z and the other random variables.

i Compute the mutual information between Z and X, i.e., $I(X,Z)$ based on the frequencies observed in the data.

ii Compute the mutual information between Z and Y, i.e., $I(Y,Z)$ based on the frequencies observed in the data.

(b) Based on your observations in part (a), which feature should be selected as candidate parent for Z? Why?
(c) In the first round of the algorithm, suppose that we choose Y to be the parent of Z in our network, X to be the parent of Y, and that X remains parent-less. Estimate the parameters of the current Bayes net, given the data.
(a)
(i) $I(X,Z)=\sum_{x}^{X}\sum_{z}^{Z}P(x,z)log_2\frac{P(x,z)}{P(x)P(z)}$
$P(x=T)=0.5$, $P(x=F)=0.5$, $P(z=T)=0.55$, $P(z=F)=0.45$
$P(x=T,z=T)=0.38$, $P(x=T,z=F)=0.12$, $P(x=F,z=T)=0.17$, $P(x=F,z=F)=0.33$
So $I(X,Z)=0.1328$
(ii) Similarly as (i), $P(y=T)=0.5$, $P(y=F)=0.5$,
$P(y=T,z=T)=0.45$, $P(y=T,z=F)=0.05$, $P(y=F,z=T)=0.1$, $P(y=F,z=F)=0.4$
So $I(Y,Z)=0.3973$

(b) Feature Y would be chosen as candidate parent for Z. Because $I(Y,Z)>I(X,Z)$

(c) There are three parameters in current Bayes net, $P(X)$, $P(Y|X)$, $P(Z|Y)$.
These parameters are as follows:

$p(X)$	T	F
	0.5	0.5

$p(Y\|X)$	X	T	F
	T	0.8	0.2
	F	0.2	0.8

$p(Z\|Y)$	Y	T	F
	T	0.9	0.1
	F	0.2	0.8

10 Kernels [5 pts]

Suppose you are given the following instances in 2-D space.

X coordinate	Y coordinate
12	4
3	18
6	11
5	5

Build the Kernel Matrix for the above dataset for each of these kernels. That is, compute a matrix K with entry K_{ij} being the kernel value between point i and point j.
(a) Polynomial kernel of degree 2, i.e., $k(x, z) = (x \cdot z)^2$. (b) RBF kernel with $k(x, z) = \exp(-\gamma|x_1 - x_2|^2)$ with $\gamma = 0.01$.
(a) $k(x_i, x_j) = (x_i x_i' + x_j x_j')^2$
So Kernel Matrix is

$$\begin{bmatrix} 160 & 108 & 116 & 80 \\ 108 & 333 & 216 & 105 \\ 116 & 216 & 157 & 85 \\ 80 & 105 & 85 & 50 \end{bmatrix}$$

(b) $k(x_i, x_j) = \exp(-0.01|x_i - x_j|^2)$
So Kernel Matrix is

$$\begin{bmatrix} 1.0 & 0.0627 & 0.427 & 0.607 \\ 0.0627 & 1.0 & 0.560 & 0.177 \\ 0.427 & 0.560 & 1.0 & 0.691 \\ 0.607 & 0.177 & 0.691 & 1.0 \end{bmatrix}$$

11 Kernel Methods [10 points]

Consider the following kernel

$$k(z, z') = \begin{cases} 1 & \text{if } \|z - z'\|_2 \leq 1, \\ 0 & \text{otherwise.} \end{cases}$$

Given data set $x_1 = (0, 0), y_1 = 1, \quad x_2 = (0, 1), y_2 = 2, \quad x_3 = (1, 0), y_3 = 3$, define function $f(x) = \sum_{i=1}^{3} \alpha_i y_i k(x, x_i)$ where the coefficients $\alpha_i = i$. Compute $f(x)$ for $x = (1, 1)$.
For $x = (1, 1)$, $k(x, x_1) = 0$, $k(x, x_2) = 1$, $k(x, x_3) = 1$
$f(x) = 0 + 2 \times 2 + 3 \times 3 = 13$

12 Principal Component Analysis [10 points]

What is the first principal component of the following data points:

$$x_1 = (-1, 0), x_2 = (1, 0), x_3 = (0, -0.1), x_4 = (0, 0.1).$$

The correlation matrix is $XX^T =$

$$\begin{bmatrix} 2.0 & 0 \\ 0 & 0.2 \end{bmatrix}$$

The eigenvalues W are

$$\begin{bmatrix} 2 & 0.02 \end{bmatrix}$$

The first principal component v has $(XX^T)v = \lambda v$, it is the eigenvector of XX^T associated with the largest eigenvalue 2
So the first principal component $v =$

$$\begin{bmatrix} 1 & 0 \end{bmatrix}$$

13 Reinforcement Learning [10 points]

Consider the deterministic reinforcement environment drawn below (let $\gamma = 0.1$). the number on the arcs indicate the immediate rewards. Assume we learn a Q-table. Also assume all the initial values in your Q table are 5.

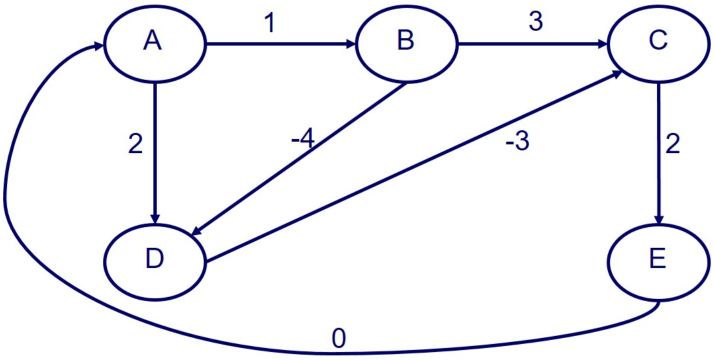


Figure 2: A deterministic reinforcement environment.

Suppose the learner follows the path $A \rightarrow D \rightarrow C \rightarrow E \rightarrow A$. Using the standard Q learning for deterministic reinforcement environment, report the final Q table on the graph above.
The final Q table is as follows:

	A	B	C	D	E
A	5	5	5	2.5	5
B	5	5	5	5	5
C	5	5	5	5	2.5
D	5	5	-2.5	5	5
E	0.5	5	5	5	5

Part 2: Extra Credits

14 Decision Tree Rank [5 points]

The rank of a decision tree is defined as follows. If the tree is a single leaf then the rank is 0. Otherwise, let r_L and r_R be the ranks of the left and right subtrees of the root, respectively. If $r_L = r_R$ then the rank of the tree is $r_L + 1$. Otherwise, the rank is the maximum of r_L and r_R. Prove that a decision tree with n leaves has rank at most $\log_2(n)$.
i For a decision tree with single leaf, obviously it is true: $\log_2(1) = 0$, and the rank is 0
ii For a decision tree with more than one leaf, assume the rank of root is r_{root}, r_L and r_R be the ranks of the left and right subtrees of the root. So the number of left subtree and right subtree is $2^{r_L} + 2^{r_R}$, which is at least $2^{r_{root}}$

15 Kernel Methods [10 points]

Car-talk statistician Marge Innovera proposes the following simple kernel function:

$$k(z, z') = \begin{cases} 1 & \text{if } z = z', \\ 0 & \text{otherwise.} \end{cases}$$

Marge likes this kernel because in the Φ-space, any labeling of the points in the instance space X will be linearly separable. So, this should be perfect for learning any target function you want to: just run a kernelized version of SVM.
1) Why is any assignment of labels to points linearly separable?
2) Nonetheless, what is the problem with her reasoning?

1) In the Φ-space, the kernel matrix has $x_{ij} = 1$ for $x_i = x_j$, and the other elements are 0. Therefore, there exist a vector w that $w^T x_{ij} + w_0 > 0$ with $w_0 > 0$ for all i, j. So any assignment of labels to points linearly separable.
2) In the test case, the result is always 0.

16 Support Vector Machines [10 points]

Given data $\{(x_i, y_i), 1 \le i \le n\}$, the (hard margin) SVM objective is

$$\min_{w,b} \frac{1}{2}\|w\|_2^2$$
$$\text{s.t. } y_i(w^\top x_i + b) \ge 1 (\forall i).$$

The dual is

$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^\top x_j$$
$$\text{s.t. } \alpha_i \ge 0 (\forall i), \quad \sum_{i=1}^{n} \alpha_i y_i = 0.$$

Suppose the optimal solution for the dual is $\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_n^*)$, and the optimal solution for the primal is (w^*, b^*). Show that the margin

$$\gamma = \min_i \frac{y_i((w^*)^\top x_i + b^*)}{\|w^*\|_2}$$

satisfies

$$\frac{1}{\gamma^2} = \sum_{i=1}^{n} \alpha_i^*.$$

Hint: use the KKT conditions.
The support vectors have $\gamma = \frac{1}{\|w^*\|_2}$,
For dual problem, according to KTT conditions we have
$w^* = \sum\limits_{i=1}^{n} \alpha_i^* x_i y_i$
$\sum\limits_{i=1}^{n} \alpha_i^* y_i = 0$
$\alpha_i^*(1 - y_i((w^*)^\top x_i + b^*)) = 0$
Therefore, $\|w^*\|_2 = \sum\limits_{i=1}^{n} \alpha_i^* y_i((w^*)^\top x_i + b^*) = \sum_{i=1}^{n} \alpha_i^*$
So $\frac{1}{\gamma^2} = \sum_{i=1}^{n} \alpha_i^*$.