Name: ________________________________ NetID: ____________

S&DS 365 / 665

Intermediate Machine Learning

Midterm Exam (Sample Solution)

October 14, 2024

Complete all of the problems. You have 75 minutes to complete the exam.

The exam is closed book, computer, phone, etc. You are allowed one double-sided $8\frac{1}{2} \times 11$ sheet of paper with hand-written notes. No calculators—one problem requires some multiplication and addition that you can do on paper.

The following facts may (or may not) be helpful:

- If (X_1, X_2) are jointly Gaussian with distribution
$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} A & C \\ C^T & B \end{pmatrix} \right)$$
then the conditional distributions are also Gaussian and given by
$$\begin{aligned} X_1 \,|\, x_2 &\sim N\left(\mu_1 + CB^{-1}(x_2 - \mu_2),\ A - CB^{-1}C^T\right) \\ X_2 \,|\, x_1 &\sim N\left(\mu_2 + C^T A^{-1}(x_1 - \mu_1),\ B - C^T A^{-1} C\right) \end{aligned}$$
- The function `numpy.random.choice(a, p)` returns a random sample from a given array `a`, with weights `p`.
- The function `numpy.linalg.inv(A)` computes the inverse of a matrix `A`
- The identity function is $\mathrm{id}(u) = u$. The rectified linear unit activation function is defined by $\texttt{relu}(u) = \max(u, 0)$ and the hyperbolic tangent activation function is $\tanh(u) = \dfrac{e^u - e^{-u}}{e^u + e^{-u}} = 2\sigma(2u) - 1$ where $\sigma(u) = \dfrac{1}{1 + e^{-u}}$ is the sigmoid.
- If $X \sim \text{Beta}(\alpha, \beta)$ is a Beta-distributed random variable, the mean and variance of X are given by
$$\mathbb{E}(X) = \frac{\alpha}{\alpha + \beta}$$
$$\text{Var}(X) = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}$$
- The multivariate Gaussian density is
$$p(x; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu) \right).$$

1. ***Multinomial choice*** (11 points)

For each of the following questions, circle the *single best* answer, unless the question allows for multiple answers.

1.1. Consider the toy lasso and ridge problem

$$\widehat{\beta} = \arg\min_{\beta} \left\{ (Y - \beta)^2 + \beta^2 + |\beta| \right\}$$

where Y is a random variable and β is a scalar. If $Y = -1$ the solution is

(a) $\widehat{\beta} = 0$

(b) $\widehat{\beta} = -\frac{1}{2}$

[✓ (c)] $\widehat{\beta} = -\frac{1}{4}$

(d) $\widehat{\beta} = \frac{1}{4}$

(e) None of the above

1.2. Suppose that we have a kernel regression technique in one dimension with bandwidth parameter h for which the squared bias scales as $O(h^3)$ and the variance scales as $O\left(\frac{1}{nh^3}\right)$ as $h \to 0$ with $nh^3 \to \infty$, for a sample of size n, under certain assumptions. What is the fastest rate at which the risk (expected squared error) will decrease with sample size for this technique?

(a) $O(n^{-1/3})$

(b) $O(n^{-1/4})$

[✓ (c)] $O(1/\sqrt{n})$

(d) $O(1/n)$

(e) None of the above

1.3. Which of the following properties are satisfied by the minimum norm solution $\widehat{\beta}$ defined for an $n \times p$ design matrix $\mathbb{X}$ with rank $n < p$ and a response vector $Y \in \mathbb{R}^n$? Circle all that apply.

[✓ (a)] $\mathbb{X}\widehat{\beta} = Y$

[✓ (b)] $\widehat{\beta} = \lim_{\lambda \to 0} \mathbb{X}^T(\mathbb{X}\mathbb{X}^T + \lambda I)^{-1}Y$

[✓ (c)] $\widehat{\beta} = \lim_{\lambda \to 0}(\mathbb{X}^T\mathbb{X} + \lambda I)^{-1}\mathbb{X}^T Y$

[✓ (d)] $\widehat{\beta}$ is a linear combination of the rows of $\mathbb{X}$.

(e) $\|\widehat{\beta}\| \leq \|\widehat{\beta}_\lambda\|$ where $\widehat{\beta}_\lambda = (\mathbb{X}^T\mathbb{X} + \lambda I)^{-1}\mathbb{X}^T Y$ for any $\lambda > 0$.

1.4. Which of the following could be Gram matrices $\mathbb{K} = [K(x_i, x_j)]$ for some Mercer kernel K? Circle your answers.

(a) $\mathbb{K} = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$ not positive definite

(b) $\mathbb{K} = \begin{pmatrix} 2 & 1 \\ -1 & 2 \end{pmatrix}$ not symmetric

$\boxed{\text{[c]}}$ $\mathbb{K} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$ ✓

$\boxed{\text{[d]}}$ $\mathbb{K} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$ ✓

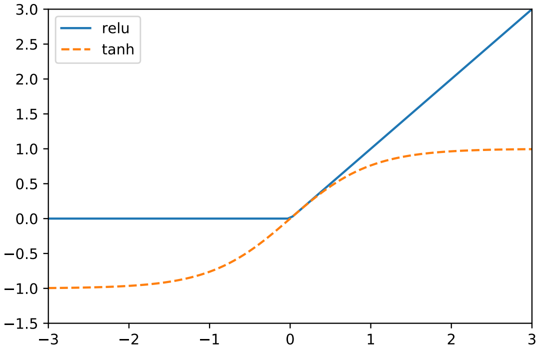
(e) $\mathbb{K} = \begin{pmatrix} -2 & 1 \\ 1 & -2 \end{pmatrix}$ diagonals not positive

1.5. Consider a neural network for binary classification of inputs $x = (x_1, x_2, x_3)^T$. The neural network defines a classifier with discriminant function

$$\log\left(\frac{p(Y = 1 \,|\, x)}{p(Y = 0 \,|\, x)}\right) = \beta^T h(x)$$

with $h(x) = \varphi(Wx)$ where φ is an activation function and W is the matrix $\begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}$ and $\beta = (1, -1, 1)^T$.

Recall the relu and tanh activation functions look like this, with $\tanh(-x) = -\tanh(x)$:



Which of the following are true? Circle your answers.

(a) If φ is relu and $x = (1, 2, 1)$ then $p(Y = 1 \,|\, x) > \frac{1}{2}$

$\boxed{\text{[b]}}$ If φ is tanh and $x = (1, -2, 1)$ then $p(Y = 1 \,|\, x) > \frac{1}{2}$

(c) If φ is tanh and $x = (1, 1, 1)$ then $p(Y = 1 \,|\, x) > \frac{1}{2}$

2. ***Convoluted thinking*** (12 points)

The following TensorFlow code constructs a CNN to classify 32×32 pixel color images of cats and dogs, using a CNN with two convolutional layers with max pooling, followed by a dense layer.

```
from tensorflow.keras import layers, models
model = models.Sequential(name="question 2")
model.add(layers.Conv2D(10, (3, 3), activation='relu', input_shape=(32, 32, 3)))
model.add(layers.MaxPooling2D((3, 3)))
model.add(layers.Conv2D(10, (2, 2), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Flatten())
model.add(layers.Dense(2, activation='tanh'))
model.summary()
```

Indicate the shape of the output tensor for each layer, together with the number of trainable parameters, by filling in the two missing fields for each of the rows below. For partial credit if an answer is wrong, you may show your work below the table. No calculators.

```
Model: "question 2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 30, 30, 10)        280
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 10, 10, 10)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 9, 9, 10)          410
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 4, 4, 10)          0
_________________________________________________________________
flatten (Flatten)            (None, 160)               0
_________________________________________________________________
dense (Dense)                (None, 2)                 322
=================================================================
Total params: 1,012
```

3. ***ELBO room*** (8 points)

(1) What is the purpose of the ELBO, as used in variational approximation?

To approximate the posterior distribution over the latent variables.

(2) How is the ELBO is defined?

$\text{ELBO}(q) = H(q) + \mathbb{E}_q \log p(x, Z)$

(3) Consider the following mixture model:

$$\begin{aligned} \mathbb{P}(Z = 1) = \mathbb{P}(Z = 0) = \frac{1}{2} \\ X \mid Z = 1 \sim N(\mu_1, 1) \\ X \mid Z = 0 \sim N(\mu_0, 1) \end{aligned}$$

where $N(\mu, 1)$ is a 1-dimensional Gaussian distribution with mean μ and variance 1. Define a variational distribution and give an explicit expression for the ELBO for this model.

Define $q(Z = 1 \mid x) = q_1$ and $q(Z = 0 \mid x) = q_0$ with constraint $q_0 + q_1 = 1$. Then

$$\text{ELBO}(q_0, q_1) = -q_1 \left(\log q_1 + \frac{1}{2}(x - \mu_1)^2 - c \right) - q_0 \left(\log q_0 + \frac{1}{2}(x - \mu_0)^2 - c \right)$$

for some constant c. To see this, note that the entropy of a Bernoulli is

$$H(q) = -q \log(q) - (1 - q) \log(1 - q).$$

Also, $\mathbb{E}_q \log p(x, Z)$ is given by

$$q \left(-\frac{1}{2}(x - \mu_1)^2 + c \right) + (1 - q) \left(-\frac{1}{2}(x - \mu_0)^2 + c \right)$$

for some constant c.

(4) What variational distribution maximizes the ELBO for the above model?

Applying Bayes' rule, the true posterior is

$$q_1 = \frac{\exp\left(-\frac{1}{2}(x - \mu_1)^2\right)}{\exp\left(-\frac{1}{2}(x - \mu_1)^2\right) + \exp\left(-\frac{1}{2}(x - \mu_0)^2\right)}.$$

We know (as discussed in class) that the true posterior maximizes the ELBO. Since the variational distribution above is completely general (since $Z \in \{0, 1\}$), this has to optimize the ELBO. Calculus shows that this posterior solves the gradient equation for the ELBO above.

4. ***Stochastic process of elimination*** (12 points)

We defined two fundamental stochastic processes used in nonparametric Bayesian inference: The Gaussian process and the Dirichlet process. This question tests your understanding of how these processes are defined.

(1) Suppose that $m(x)$ denotes a function with 1-dimensional input $x \in \mathbb{R}$. We place a Gaussian process prior $\pi(m)$ on such functions with mean function $\mu(x)$ and covariance kernel $K(x, x') = K_\sigma(x - x')$ where K_σ denotes a Gaussian kernel with mean zero and variance σ^2.

(a) What is the prior mean $\mathbb{E}_\pi(m(x))$ of m at a fixed point $x \in \mathbb{R}$?

By the definition of Gaussian process, $m(x) \sim N(\mu(x), K(x, x))$.
So, $\mathbb{E}_\pi(m(x)) = \mu(x)$

(b) What is the prior variance $\text{Var}_\pi(m(x))$ of m at a fixed point $x \in \mathbb{R}$?

$\text{Var}_\pi(m(x)) = K_\sigma(0) = \frac{1}{\sqrt{2\pi\sigma^2}}$

(c) Does the prior belief in $m(x)$ become stronger or weaker as $\sigma \to \infty$? Explain.

As $\sigma \to \infty$ the variance $\to 0$. The prior belief that the mean is μ becomes stronger.

(2) Suppose that $F(x)$ denotes a 1-dimensional distribution function $x \in \mathbb{R}$, so $F(x) = \mathbb{P}_F(X \leq x)$. We place a Dirichlet process prior $\pi(F)$ on such distribution functions with parameters $\alpha > 0$ and F_0 where F_0 is a fixed distribution.

(a) What is the prior mean $\mathbb{E}_\pi(F(x))$ of F at a fixed point $x \in \mathbb{R}$?

By the definition of the Dirichlet process,

$$F(x) \sim \text{Beta}(\alpha F_0(x), \alpha(1 - F_0(x))).$$

Using the expectation of the Beta on the cover page, we get $\mathbb{E}_\pi(F(x)) = F_0(x)$

(b) What is the prior variance $\text{Var}_\pi(F(x))$ of F at a fixed point $x \in \mathbb{R}$? (Hint: See the cover page of the exam.)

$$\text{Var}_\pi(F(x)) = \frac{F_0(x)(1 - F_0(x))}{(1 + \alpha)}$$

(c) Does the prior belief in $F(x)$ become stronger or weaker as $\alpha \to \infty$? Explain.

As $\alpha \to \infty$ the variance $\to 0$. The prior belief that the mean is F_0 becomes stronger.

5. ***Color me mine*** (15 points)

For this problem, we extend the Ising model to be a collection variables $Z_{(x,y)} \in \{0, 1, 2, 3\}$ on a 30×30 grid. Each of the four possible values of $Z_{(x,y)}$ corresponds to a color: red, blue, green, or yellow.

The joint probability takes the form

$$p(Z) \;\propto\; \exp\left(\beta \sum_{(x,y)\sim(x',y')} \mathbb{1}\left(Z_{(x,y)} = Z_{(x',y')}\right)\right)$$

where β is scalar parameter. Here $(x, y) \sim (x', y')$ means that points (x, y) and (x', y') are connected in the 30×30 grid, where x and y range from 0 to 29, and $\mathbb{1}\left(Z_{(x,y)} = Z_{(x',y')}\right)$ is 1 if the two values (colors) are the same, and 0 otherwise.

Recall that the Gibbs sampling algorithm draws from this distribution by repeatedly visiting nodes (x, y), and sampling $Z_{(x,y)}$ while holding all of the other $Z_{(x',y')}$ values fixed.

(a) Suppose that we are running a Gibbs sampling step on a node (x, y) that has four neighbors, with

$$Z_{(x-1,y)} = Z_{(x+1,y)} = Z_{(x,y-1)} = 1$$
$$Z_{(x,y+1)} = 2.$$

In this Gibbs sampling step, what is the probability that $Z_{(x,y)}$ is set to 2?

(1) $1/4$

[✓ (2)] $e^{\beta}/(2 + e^{\beta} + e^{3\beta})$

(3) $e^{3\beta}/(2 + e^{\beta} + e^{3\beta})$

(4) $e^{\beta}/(1 + e^{\beta} + e^{3\beta})$

(5) None of the above

The following code partially implements Gibbs sampling for this model. Your job is to complete the implementation by providing five additional lines of code.

```
import numpy as np
num_colors = 4

def gibbs_step(Z, beta, x, y):
    # line 1
    exponent = ...

    for dx, dy in [(-1,0), (1,0), (0,-1), (0,1)]:
        if ((x + dx < 0) | (x + dx >= Z.shape[0]) |
            (y + dy < 0) | (y + dy >= Z.shape[1])):
            continue
        for c in range(num_colors):
            # line 2
            exponent[c] = ...

    # line 3
    weights = ...

    # line 4
    Z_new = ...

    return Z_new

def run_gibbs_sampling(Z, beta, steps=100):
    for _ in np.arange(steps*np.prod(Z.shape)):
        x = np.random.choice(range(Z.shape[0]))
        y = np.random.choice(range(Z.shape[1]))
        # line 5
        Z[x, y] = ...
```

(b) Complete the implementation, by writing the five missing lines below.

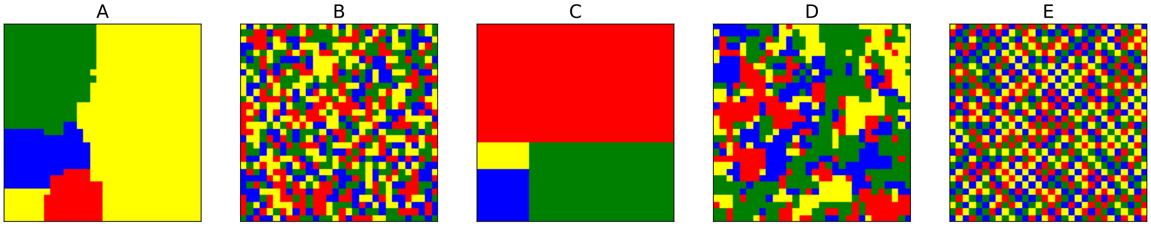
```
# line 1:
exponent = np.zeros(num_colors)

# line 2:
exponent[c] += beta * (Z[x+dx, y+dy] == c)

# line 3:
weights = np.exp(exponent) / np.sum(np.exp(exponent))

# line 4:
Z_new = np.random.choice(range(num_colors), p=weights)

# line 5:
Z[x, y] = gibbs_step(Z, beta, x, y)
```



(c) The figure above shows the result of running Gibbs sampling for 270,000 steps using five different values of β:

$$\beta \in \{-10, 0, 1, 2, 10\}$$

Which is which? Explain your answer.

$$\text{A}: \quad \beta = 2$$

$$\text{B}: \quad \beta = 0$$

$$\text{C}: \quad \beta = 10$$

$$\text{D}: \quad \beta = 1$$

$$\text{E}: \quad \beta = -10$$

6. ***Map for success*** (10 points)

In class and on the assignments, we used Gaussian processes and Mercer kernels mainly for regression. But they can also be used for classification.

For binary classification, a natural approach is to use a Gaussian process prior $m \sim GP(0, K_\sigma)$ with mean zero and covariance given by a Gaussian kernel K_σ with variance σ^2 and the likelihood model

$$\mathbb{P}(y = 1 \,|\, m, x) = \frac{e^{\beta m(x)}}{1 + e^{\beta m(x)}} = \text{sigmoid}(\beta m(x))$$

where β is a fixed scalar parameter. This is a type of nonparametric Bayesian model for classification. Unfortunately, the posterior is not a Gaussian process, and can't be easily computed. As a compromise, we can compute the MAP (maximum a posteriori) estimate.

(a) Give the loss function that the MAP estimate $\widehat{m} = \arg\max_m \mathbb{P}(m \,|\, \{x_i, y_i\})$ must minimize, with respect to a training set $\{(x_i, y_i)\}$ with $y_i \in \{0, 1\}$. Explain and justify your choice of loss function.

The MAP maximizes the prior times the likelihood; so we minimize the negative log-likelihood minus the log-prior. The negative log-likelihood of a point (x, y) for the model m is

$$-\log p(y \,|\, x, m) = \log\left(1 + \exp(\beta m(x))\right) - y\beta m(x).$$

The negative log prior is

$$\frac{1}{2}\alpha^T \mathbb{K}\alpha + \text{const}$$

when $m = \mathbb{K}\alpha$ is evaluated on data points $(x_1, \ldots, x_n)$. By the representer theorem, it suffices to restrict to such m. Putting the terms together, in vector notation, the loss function is

$$\mathbb{1}^T \log\left(1 + \exp(\beta\mathbb{K}\alpha)\right) - \beta y^T \mathbb{K}\alpha + \frac{1}{2}\alpha^T \mathbb{K}\alpha$$

(b) Give a stochastic gradient descent algorithm for estimating a function m that minimizes this loss function. For full credit, give the algorithm in an explicit form that could be implemented.

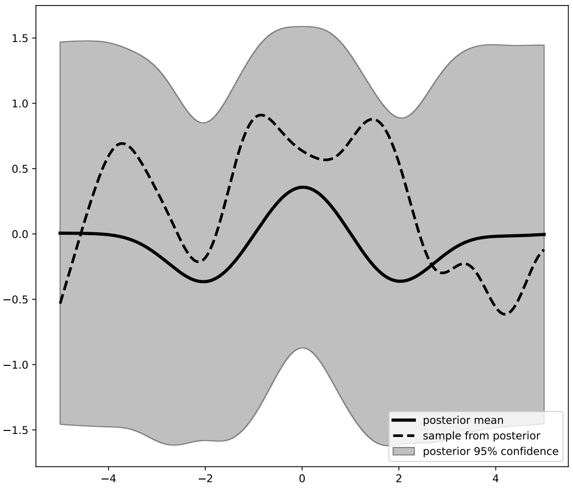
Using the loss function above, and taking gradients, the gradient descent algorithm is

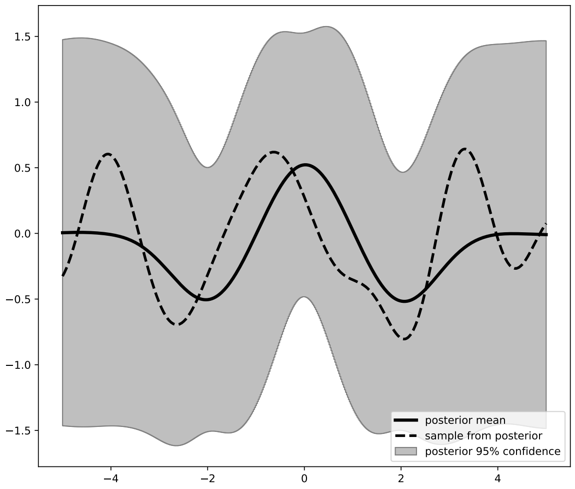
$$p = \exp(\beta\mathbb{K}\alpha)/(1 + \exp(\beta\mathbb{K}\alpha))$$

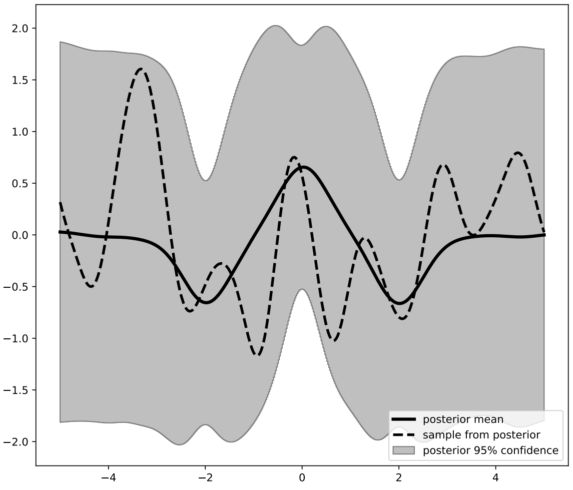
$$\alpha \leftarrow \alpha - \eta\,(\beta\mathbb{K}(p - y) + \mathbb{K}\alpha)$$

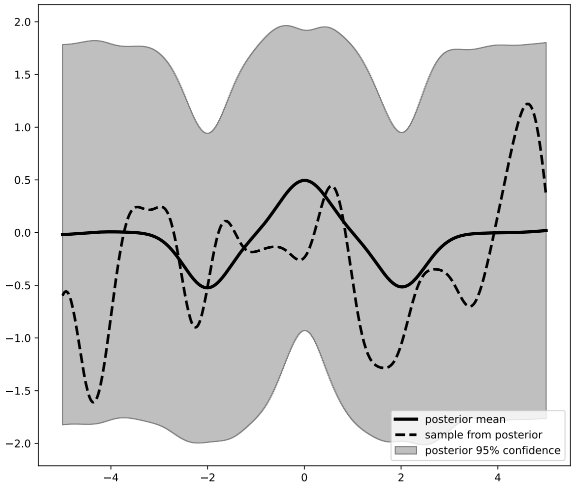
Restricting to a subset of points gives a minibatch SGD algorithm.

The following four plots illustrate posterior inference for this Gaussian process classification model with different settings of the Gaussian kernel variance σ^2 and the likelihood scaling parameter β. The three training data points (x, y) were $\{(-2, 0), (0, 1), (2, 0)\}$.


Plot A


Plot B


Plot C


Plot D

(d) Each plot was generated with a choice of β and σ, chosen from two possibilities:

$$\sigma \in \left\{\tfrac{1}{2}, \tfrac{3}{4}\right\} \quad \beta \in \{2, 5\}$$

Indicate what the parameters were for each plot:

Plot A: $\sigma = \frac{3}{4} \quad \beta = 2$

Plot B: $\sigma = \frac{3}{4} \quad \beta = 5$

Plot C: $\sigma = \frac{1}{2} \quad \beta = 5$

Plot D: $\sigma = \frac{1}{2} \quad \beta = 2$