

## 1. MLE (35 points)

You have the following data:

$$\text{data:}(t_i, x_i) : i = 1, \dots, N$$

where  $t_i$  measures the duration of an unemployment spell, and  $x_i$  is a  $K \times 1$  vector of observable characteristics. You plot the empirical CDF of the data and believe that the durations follow an exponential distribution. You assume that the density of durations is given by:

$$f(t|x, \beta) = \lambda(x; \beta) e^{-\lambda(x; \beta)t}$$

where  $\ln \lambda(x; \beta) = x' \beta$ , where  $\beta$  is  $K \times 1$  vector of parameters.

1. (10 points) Write down the log likelihood, and its first order conditions to find expressions for  $\beta$ 's

2. (5 points) Assume that:

$$x_i \sim N(0, 1)$$

$$\beta = \begin{pmatrix} 1 \\ 4 \end{pmatrix}$$

Write a Julia program that creates a simulated data set with the information above for  $N=100$  individuals.

HINT: to create  $t_i$ , please use the following formula

3. (10 points) Using the optimizers, find MLE estimates. Verify that they are close to the true value for  $\beta$ .
4. (10 points) Empirically show consistency and asymptotic normality of beta parameter.

Note: this may take time, so solve other parts first to save some time.

## 2. GMM (35 points)

Consider the model

$$y_i = x_i' \beta + \epsilon_i$$

$$E(z_i \epsilon_i) = 0$$

$$R' \beta = 0$$

with  $y_i$  scalar,  $x_i$  a  $k$  vector and  $z_i$  an  $l$  vector with  $l > k$ . The matrix  $R$  is  $k \times q$  with  $1 \leq q < k$ . You have a random sample  $(y_i, x_i, z_i : i = 1, \dots, n)$ .

Assume the efficient weighting matrix  $W = \Omega^{-1} = \left( E(z_i z_i' \epsilon_i^2) \right)^{-1}$ .

1. (10 points) Write out the GMM estimator  $\hat{\beta}$  of  $\beta$  given the moment condition  $E(z_i \epsilon_i) = 0$  but ignoring  $R' \beta = 0$ .

2. (10 points) Write out the GMM estimator  $\tilde{\beta}$  of  $\beta$  given the moment condition  $R' \beta = 0$  and  $E(z_i \epsilon_i) = 0$ .

Hint: The objective function should use the Lagrangian:

$$Q_n(\beta, \lambda) = \frac{1}{2} (Y - X\beta)' Z \Omega^{-1} Z' (Y - X\beta) + \lambda' R' \beta$$

where  $\lambda$  is  $q \times 1$  Lagrange multiplier. You are welcome to Google for properties of matrix algebra, of course :)

3. (10 points) Find the asymptotic distribution  $\sqrt{n}(\tilde{\beta} - \beta)$  as  $n \rightarrow \infty$  under the assumption that  $E(z_i \epsilon_i) = 0$  and  $R' \beta = 0$  are correct.
4. (5 points) Sketch how you would code this problem if you were to have the the data:
- $$\text{data:}(y_i, x_i, z_i) : i = 1, \dots, N$$
- where  $y_i$  scalar,  $x_i$  a  $k$  vector and  $z_i$  an  $l$  vector with  $l > k$ . The matrix  $R$  is  $k \times q$  with  $1 \leq q < k$ .

### 3. Nonparametrics (30 points)

Load the following data:

```
In [140]: 1 data = DataFrame(CSV.File("data1.csv"))
          2 y = data[:,1]
          3 x = data[:,2]
          4 ydata = convert(Array,y)
          5 xdata = convert(Array,x)
          6 ;
```

1. (10 points) using Nadaraya-Watson estimator, evaluate  $\hat{f}(x)$  at the following points of x.

```
In [130]: 1 collect(range(0,1,step=0.01));
```

Hint: the code given in class is written for multivariate  $X$ . To adapt to this situation where  $x_i$  is one dimension, you can just erase repeat() functions.

2. (10 points) In a separate window, (1) first scatter plot xdata and ydata, and then in the same window (2) plot  $\hat{f}(x)$ .

Hint: you can add to the previous plot by using "plot!" function. Moreover, if you cannot see the second plot/line, increase the thickness of the line of the second plot by using linewidth option (lw=5).

Looking at the graph, how does the estimated  $\hat{f}(x)$  at different  $x$  values different? In other words, can you present your  $\hat{f}(x)$  at each  $x$  values with same confidence? How would you present this finding? Please **discuss** how you would present this result.

3. (10 points) Notice that you have used Silverman's rule for picking the hyperparameter "gamma" in the notes. Estimate and plot  $\hat{f}(x)$  using different gamma's (within reason - with the xdata, gamma is 0.0619, so maybe try gamma = 0.5 and gamma = 0.001). Please **discuss** the bias and variance tradeoff given different gamma value. If the gamma is too high, how does it increase the risk; if gamma is too low, how does it increase the risk?

In [210]:

```
1 gamma = 1.06.*std(xdata,dims=1).*(size(xdata,1)^-0.2)
```

Out[210]: 1-element Array{Float64,1}:  
0.06194252878418682