Econometrics Test Exercise 2

a) Prove:
$$E(b_r) = \beta + \beta_2$$

Expressing b_R in term of

$$b_R = (x_1'x_1)^T x_1' y$$

$$= (x_1'x_1)^T x_1' (x_1\beta_1 + x_2\beta_2 + e)$$

$$b_R = (x_1'x_1)^T x_1' (x_1\beta_1 + x_2\beta_2 + e)$$

$$b_R = (x_1'x_1)^T x_1' x_1 + (x_1'x_1)^T x_1' x_2 + e$$

$$= (x_1'x_1)^T x_1' x_1 + (x_1'x_1)^T x_1' x_2 + e$$

$$= (x_1'x_1)^T x_1' x_1 + (x_1'x_1)^T x_1' x_2 + e$$

$$= x_1 + (x_1'x_1)^T x_1' x_2 + e$$

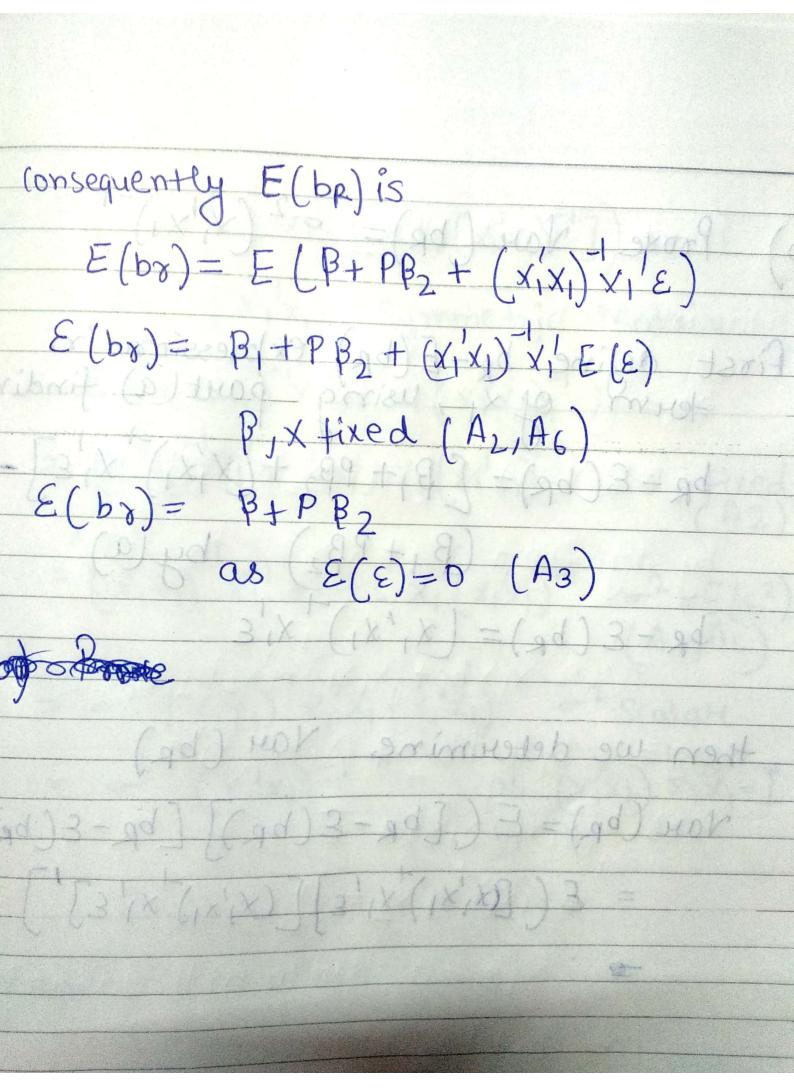
$$= x_1 + x_1' x_1 + e$$

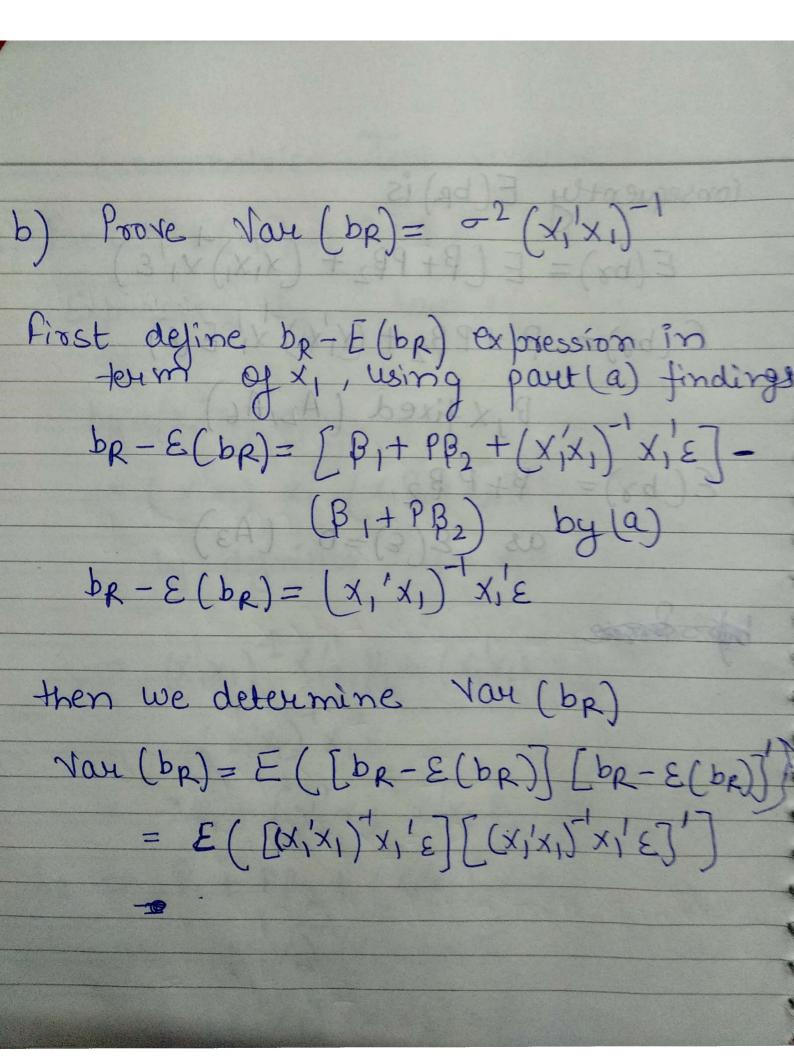
$$= x_1 + x_1' x_1 + e$$

$$= x_1 + x_1' x_1 + e$$

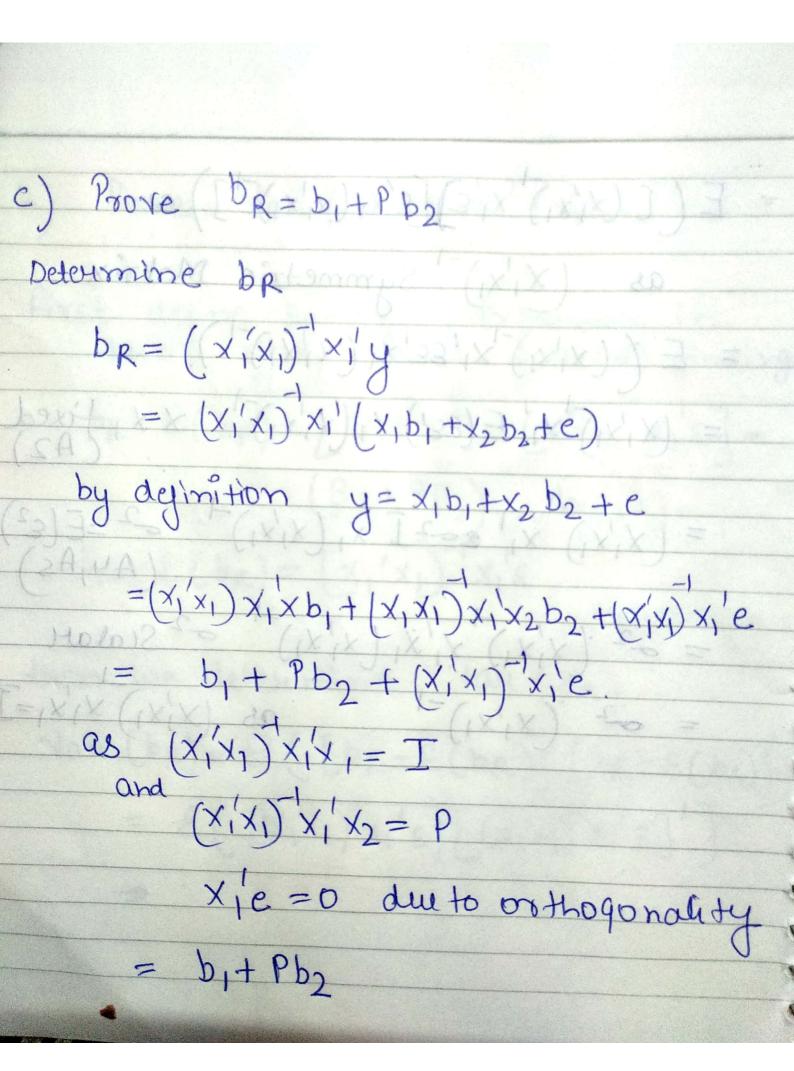
$$= x_1' x_1 + x_2' x_2 + e$$

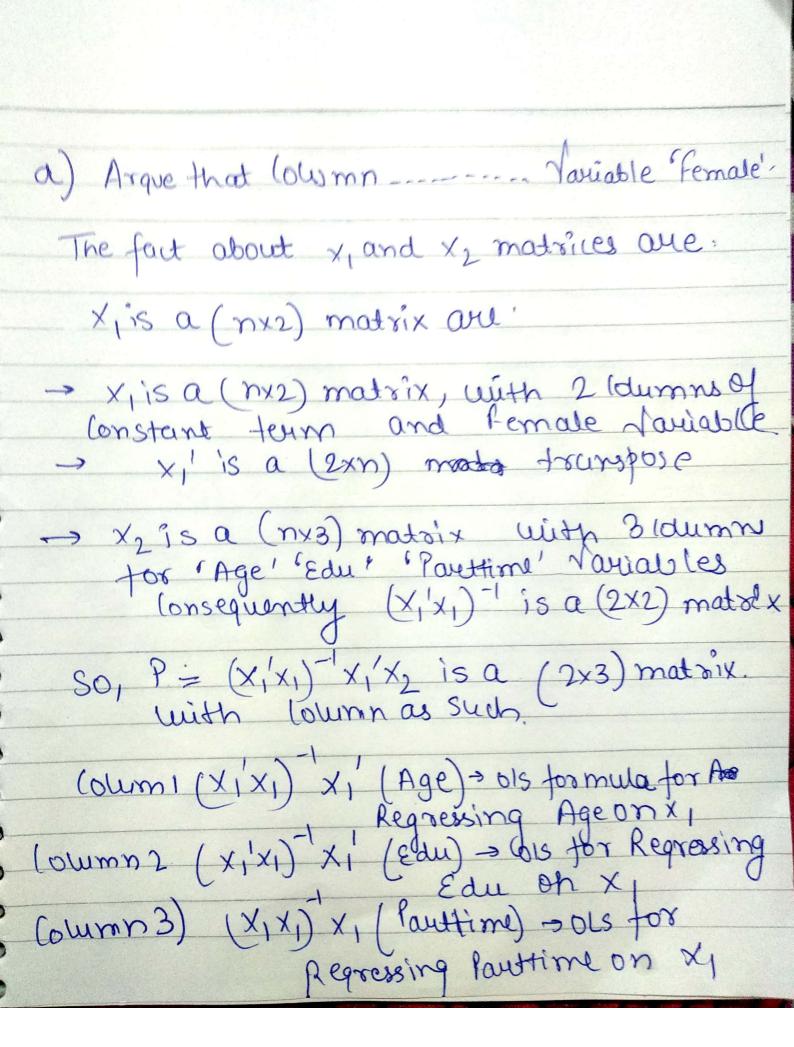
$$= x_1'$$





= E([(x/x))x's][s'x,(x/x)]) a) = as (X/X) Symmetric Matrix = E((x/x))x/23/x(x/x)) = = = $(X_1'X_1)^TX_1' \in (EE') \times ((X_1'X_1)^T)^T \times (EE') = (EE') \times (EE') \times (EE') \times (EE') = (EE')$ = $[x_1x_1]^{-1}x_1$ &= $[x_1(x_1x_1)^{-1}]^{-1}$ = $[x_1(x_1)^{-1}]^{-1}$ = $[x_1(x_1)^{-1}]^{$ = 02 (x,x,) x, x, (x,x,) -1 02 Scalare $= 3^2 (x_1 x_1)^{-1}$ etiloraportro et ub 0 = six





Week2

September 9, 2018

```
In [4]: # getting the necessary libraries to read the data
        import pandas as pd
        import numpy as np
        # import the libraries for any potential mathematical operation
        import math
        # getting libraries for the scatter diagram
        import matplotlib.pyplot as plt
        import seaborn as sns
        # necessary function to display the chart here once it is generated
       %matplotlib inline
In [5]: testExer2=pd.read_excel("C:\Users\\vishe\Downloads\Dataset2.xls",sheet_name='Dataset L
       testExer2.head()
Out[5]:
          Observation Wage
                             LogWage Female Age Educ Parttime
       0
                         66 4.189655
                                                49
                                                       1
                    1
                                            0
                                                                 1
       1
                         34 3.526361
                                            1
                                                42
                                                       1
                                                                 1
                    3
                         70 4.248495
                                            1 42
                                                       1
                         47 3.850148
                                                38
                                                       1
                                                                 0
                        107 4.672829
In [7]: #creating a constant term variable of value=1
       testExer2['Constant']=1
       testExer2.head()
                                                    Educ Parttime Constant
Out[7]:
          Observation Wage
                             LogWage
                                      Female
                                               Age
       0
                         66 4.189655
                                                49
                    1
                                            0
                                                       1
                                                                           1
       1
                    2
                         34 3.526361
                                            1
                                              42
                                                       1
                                                                 1
                                                                           1
                    3
                         70 4.248495
                                            1 42
                                                       1
                                                                 1
                                                                           1
       3
                         47 3.850148
                                                38
                                                       1
                                                                 0
                                                                           1
                        107 4.672829
                                                54
```

Building X1 (nx2) matrix

```
In [ ]: X1=testExer2.as_matrix(columns={'Constant', 'Female'})
```

```
and its X1' (2×n) transpose...
In [13]: X1transpose= X1.transpose()
          Building X2 (n×3) matrix...
 In [ ]: X2=testExer2.as_matrix(columns={'Age', 'Educ', 'Parttime'})
          Producing X1'X1 (2×2) matrix...
In [21]: X3=np.matmul(X1transpose,X1)
          ... and its (X1'X1)^-1 (2×2) inverse...
In [30]: X4=np.linalg.inv(X3)
Out[30]: array([[ 0.00316456, -0.00316456],
                 [-0.00316456, 0.00859934]])
          ... then, the (X1'X1)^-1 *X'1 (2×n) matrix...
In [25]: X5=np.matmul(X4,X1transpose)
```



(f) Part (c) numerical validity checking.

or three decimals; preciser results would have been obtained for higher precision coefficients. In Lecture 2.5, the OLS regression equation was computed as:

(f) Check the numerical validity of the result in part (c). Note: This equation will not hold exactly because the coefficients have been rounded to two

$$Log(Wage)_i = 3.053 - 0.041 Female_i + 0.031 Age_i + 0.233 Educ_i - 0.365 Parttime_i + e_i$$

which determines
$$b_1=egin{bmatrix} 3.053 \\ -0.041 \end{bmatrix}$$
 and $b_2=egin{bmatrix} 0.031 \\ 0.233 \\ -0.365 \end{bmatrix}$.

b1= np.array([[3.053],[-0.041]]) In [35]:

Out[35]: array([[3.053],

```
[-0.041]])
```

```
In [36]: b2=np.array([[0.031],[0.233],[-0.365]])
Out[36]: array([[ 0.031],
```

In part (e),
$$P$$
 matrix was computed as $P=egin{bmatrix} 40.05 & 2.26 & 0.2 \\ -0.11 & -0.49 & 0.25 \end{bmatrix}$

In part (c) it was shown that: $b_R = b_1 + Pb_2$, which can be solved as:

which determined b_R as: $b_R = \begin{bmatrix} 4.73 \\ -0.25 \end{bmatrix}$.

$$egin{aligned} b_R &= b_1 + Pb_2 \ &= egin{bmatrix} 3.053 \ -0.041 \end{bmatrix} + egin{bmatrix} 40.05 & 2.26 & 0.2 \ -0.11 & -0.49 & 0.25 \end{bmatrix} egin{bmatrix} 0.031 \ 0.233 \ -0.365 \end{bmatrix} \ &= egin{bmatrix} 3.053 \ -0.041 \end{bmatrix} + egin{bmatrix} 1.695 \ -0.209 \end{bmatrix} \ &= egin{bmatrix} 4.748 \ -0.25 \end{bmatrix} \end{aligned}$$

This aligns with the b_R defined in Lecture 2.1 by the the 'Log(Wage)' OLS regression on the constant and 'Female' variable:

$$Log(Wage)_i = 4.73 - 0.25 Female_i + e_i$$

Minor b_R values difference is explained by reduced precission due to the OLS coefficients' computation rounding values.