HW3 - STAT 580 - Sp 2015

Yet T Nguyen

02/26/2015

1.

(a)

$$\begin{aligned}
P(U \le r(X)) &= \int_{\mathbb{X}} \left(\int_0^{r(x)} du \right) g(x)dx \\
&= \int_{\mathbb{X}} r(x)g(x)dx \\
&= \int_{\mathbb{X}} \frac{q(x)}{\alpha g(x)} g(x)dx \\
&= \frac{1}{\alpha} \int_{\mathbb{X}} q(x)dx.
\end{aligned}$$

(b)

$$\begin{aligned}
P(X \in A, U \le r(X)) &= \int_{A} \left(\int_0^{r(x)} du \right) g(x)dx \\
&= \int_{A} r(x)g(x)dx \\
&= \int_{A} \frac{q(x)}{\alpha g(x)} g(x)dx \\
&= \frac{1}{\alpha} \int_{A} q(x)dx.
\end{aligned}$$

Therefore,

$$\begin{aligned}
P(Y \in A) &= P(X \in A | U \le r(X)) \\
&= \frac{P(X \in A, U \le r(X))}{P(U \le r(X))} \\
&= \frac{\frac{1}{\alpha}\int_A q(x)dx}{\frac{1}{\alpha}\int_{\mathbb{X}} q(x)dx} \\
&= \frac{\frac{1}{\alpha}\int_A cq(x)dx}{\frac{1}{\alpha}\int_{\mathbb{X}} cq(x)dx} \\
&= \frac{\frac{1}{\alpha}\int_A f(x)dx}{\frac{1}{\alpha}\int_{\mathbb{X}} f(x)dx} \\
&= \int_A f(x)dx \quad (\text{since } \int_{\mathbb{X}} f(x)dx = 1)
\end{aligned}$$

2.

(a) Let $g(x) = c(2x^{\theta-1}e^{-x} + x^{\theta-1/2}e^{-x})$. Then

$$\begin{aligned}
g(x) &= c\left(2\Gamma(\theta) frac1\Gamma(\theta) e^{\theta-1}e^{-x} + \Gamma(\theta+1/2)\frac{1}{\Gamma(\theta+1/2)} e^{\theta-1/2}e^{-x}\right) \\
&= c(2\Gamma(\theta)\text{Gamma}(\theta,1) + \Gamma(\theta+1/2)\text{Gamma}(\theta+1/2,1)). \qquad (1)
\end{aligned}$$

Since $\int g dx = 1$, hence

$$c = \frac{1}{2\Gamma(\theta) + \Gamma(\theta + 1/2)}. \tag{2}$$

(b) and from (1), g is the mixture of 2 Gamma distribution Gamma$(\theta, 1)$ and Gamma$(\theta + 1/2, 1)$ with the corresponding weights

$$\frac{2\Gamma(\theta)}{2\Gamma(\theta) + \Gamma(\theta + 1/2)}, \quad \frac{\Gamma(\theta + 1/2)}{2\Gamma(\theta) + \Gamma(\theta + 1/2)}$$

(c) A procedure to sample from $g(x)$:

1. Generate $U \sim$ Unif$(0, 1)$, $X_1 \sim$ Gamma$(\theta, 1)$, $X_2 \sim$ Gamma$(\theta+1/2, 1)$ independently, where a procedure to sample from any Gamma$(., 1)$ distribution can be conducted using a rejection sampling procedure following from Theorem 3.3 (page 29 in the link and page 406 in the book) http://www.nrbook.com/devroye/Devroye_files/chapter_nine.pdf
2. If $U \le \frac{2\Gamma(\theta)}{2\Gamma(\theta)+\Gamma(\theta+1/2)}$ then $X = X_1$, otherwise, $X = X_2$
3. The random variable X has distribution $g(x)$.

(d) Sampling a random variable having distribution f using g as proposed distribution. It is easy to see

$$\sqrt{4 + x} \le 2 + x^{1/2},$$

hence

$$q(x) \equiv \sqrt{4 + x}x^{\theta-1}e^{-x} \le (2 + x^{1/2})x^{\theta-1}e^{-x} \le \frac{1}{c}g(x), \tag{3}$$

where c is defined as in (2).

1. Generate $X \sim g(x)$ as in part (c), and generate $U \sim$ Unif$(0, 1)$ independently. Set $r(x) = \frac{cq(x)}{g(x)}$, where q is defined as in (3).
2. If $U \le r(X)$, then accept X, otherwise, repeat step 1.

3.

```
#include <stdio.h>
#define N  16 /* number of observations */
#define P  2 /* number of predictors */

void dgesv_(int *NN, int *NRHS, double *A, int *LDA, int *IPIV,
            double *B, int *LDB, int *INFO);
int main(){
  /* longley dataset from R: Employed (Y) GNP.deflator and Population (X) */
    double Y[N] = {60.323,61.122,60.171,61.187,63.221,63.639,64.989,
                   63.761,66.019,67.857,68.169,66.513,68.655,69.564,
                   69.331,70.551};
double X[N][P] =
{{83,107.608},
{88.5,108.632},
{88.2,109.773},
{89.5,110.929},
{96.2,112.075},
{98.1,113.27},
{99,115.094},
{100,116.219},
{101.2,117.388},
{104.6,118.734},
{108.4,120.445},
{110.8,121.95},
{112.6,123.366},
{114.2,125.368},
{115.7,127.852},
```

```
{116.9,130.081}};

double XtX[(P+1)*(P+1)];
double XtY[P+1];

int ipiv[P+1];
int i, j, k, n1, n2, info;
/* Calculate (1, X)'*(1, X) which is a 3X3 matrix*/
XtX[0] = N;

for(i = 1;i<P+1;i++){
  XtX[i*(P+1) + 0] = 0;
  for(j=0;j<N;j++)
  XtX[i*(P+1) + 0] += X[j][i-1];
  XtX[0*(P+1)+i] = XtX[i*(P+1) + 0];
}

for(i=1;i<P+1;i++){
  for(j=1;j<P+1;j++){
    XtX[i*(P+1)+j] = 0;
    for(k=0;k<N;k++){
      XtX[i*(P+1)+j]+=X[k][i-1]*X[k][j-1];
    }
  }
}

/* Calculate (1,X)'*Y which is a 3x1 matrix*/
XtY[0] = 0;
for(i=0;i<N;i++)
XtY[0] +=Y[i];

for(i=1;i<P+1;i++){
  XtY[i] = 0;
  for(j=0;j<N;j++)
  XtY[i] +=X[j][i-1]*Y[j];
}
n1 = P+1;
n2 = 1;
dgesv_(&n1, &n2, XtX, &n1, ipiv, XtY, &n1, &info);

if (info != 0)
printf("dgesv error %d\n", info);
for (i=0; i<P+1; i++)
printf("%f\t", XtY[i]);
printf("\n");
return 0;
}
/* gcc -pedantic -Wall -ansi  hw33.c -llapack -lblas -lgfortran */
/* 26.851352  0.240842  0.119026*/
```

4.

(a) and (b)

```
#include <stdio.h>
#define N 10
int main()
{
  int c, d;
  double x[N] = {3.1, -1.2, 5.3, 1, 4.4, 21, 3, 7, -1.2, 3.2};
```

```
  double t;
  for (c = 1 ; c <= N - 1; c++) {
    d = c;

    while ( d > 0 && x[d] < x[d-1]) {
      t          = x[d];
      x[d]   = x[d-1];
      x[d-1] = t;

      d--;
    }
  }

  printf("Sorted data:\n");

  for (c = 0; c <= N - 1; c++) {
    printf("%f ", x[c]);
  }
 printf("\n\nMedian of x is %f \n", (x[N/2-1] + x[N/2])/2);
  return 0;
}
/*gcc -pedantic -Wall -ansi  hw34.c -llapack -lblas -lgfortran */
```

Output

```
Sorted data:
-1.200000 -1.200000 1.000000 3.000000 3.100000 3.200000 4.400000 5.300000 7.000000 21.000000

Median of x is 3.150000
```