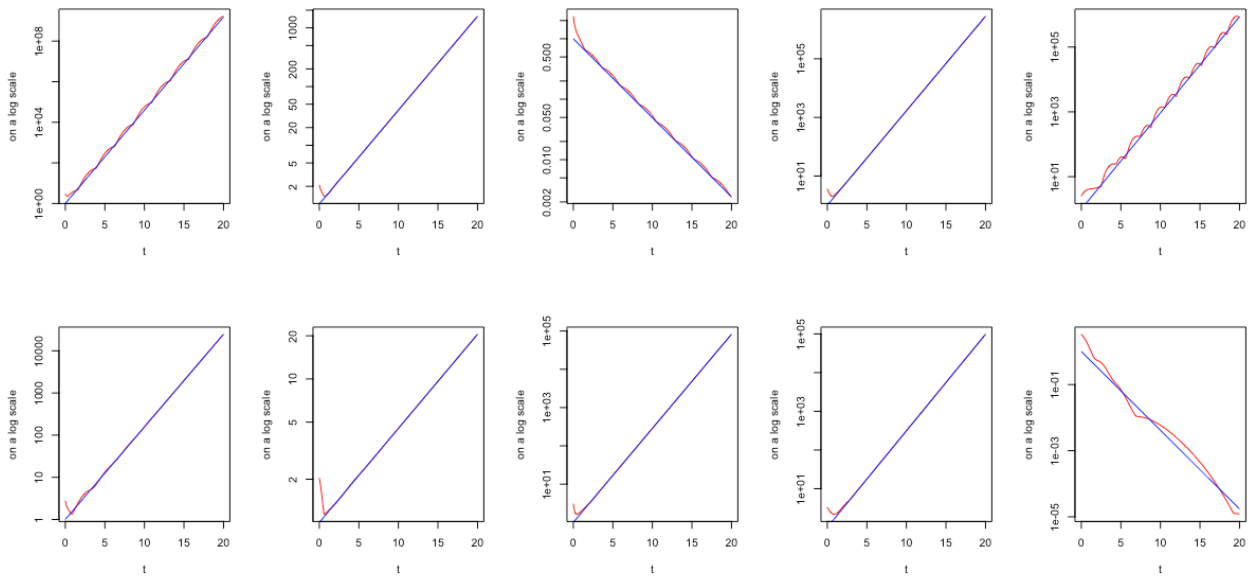
一些作业答案更正：

hw6 第二题

We try to find the relationships among $\left\|e^{tA}\right\|_2, e^{t\alpha(A)}$ and $t \in [0, 20]$, where $\alpha(A)$ is the spectral abscissa of A. We assume the eigenvalue decomposition of A is $A = X\Lambda X^T$, where $\Lambda = \text{diag}(\Lambda_1, \cdots, \Lambda_{10})$ Then,
$e^{tA} = \sum_{n=0}^{\infty} \frac{(tA)^n}{n!} = \sum_{n=0}^{\infty} \frac{t^n (X\Lambda X^T)^n}{n!} = X \sum_{n=0}^{\infty} \frac{t^n \Lambda^n}{n!} X^T = X \,\text{diag}\left(e^{t\Lambda_1}, \cdots, e^{t\Lambda_n}\right) X^T$ We generate 10 random matrices A and the results are shown below.

```r
par(mfrow = c(2,5))
for(num in 1:10){
  # Generate a Pseudo Matrix A
  A <- matrix(rnorm(100),10,10)-2*diag(10)
  # Eigenvalue Decomposition
  ev <- eigen(A)
  X <- ev$vectors
  # Method 1: exp(tA) 2-norm
  y1 <- vector()
  for (i in seq(0,20,0.1)){
    y1 <- c(y1,norm(X%*%(diag(exp(i*ev$val)))%*%t(X),type='2'))
  }
  # Method 2: exp(ta(A))
  t <- seq(0,20,0.1)
  # Get the Spectral Abscissa of A
  alpha <- max(Re(ev$val))
  y2 <- exp(t*alpha)
  plot(t,y1,col=2,type="l",xlab = "t",ylab = "on a log scale",log = "y")
  lines(t,y2,col=4,type="l")
}
```



(Note: The red curve is $\left\|e^{tA}\right\|_2$ against t and the blue line is $e^{t\alpha(A)}$ against t. And both of them are shown in a log scale.)

We find that the red curve is very close to the straight blue line in most cases. However, there are a few cases that the red curve fluctuates near the straight blue line. Besides, the red curve generally converges to the straight blue line and the oscillation fades away as t increases. If the matrix A is highly ill-conditioned, then a great deal of information may be discarded in passing from A to Λ. Besides, if A is a defective matrix, it even does not have an eigenvalue decomposition. Thus, the derivation of $\left\|e^{tA}\right\|_2$ would be distorted. Under these situations, the red curve would remains oscillatory as $t \to \infty$.

hw7 第一题（3）fisher scoring

Noted that the independent and identical distribution I of the sample, the amount of Fisher information under a single sample $I_1(\theta)$ can be calculated.

$$l_1'(\theta; x) = \frac{2(x-\theta)}{1+(x-\theta)^2}$$

Therefore,

$$\begin{aligned} I_1(\theta) = E\left[l_1'(\theta; X)^2\right] &= \frac{4}{\pi}\int_{-\infty}^{+\infty} \frac{(x-\theta)^2}{\left(1+(x-\theta)^2\right)^3} dx \\ &= \frac{4}{\pi}\int_{-\pi/2}^{\pi/2} \frac{\tan^2 t}{\left(1+\tan^2 t\right)^3}\frac{dt}{\cos^2 t} = \frac{4}{\pi}\int_{-\pi/2}^{\pi/2} \cos^2 t \sin^2 t dt \\ &= \frac{1}{\pi}\int_{-\pi/2}^{\pi/2} \sin^2 2t dt = \frac{1}{\pi}\int_0^{\pi} \sin^2 t dt = \frac{1}{2} \end{aligned}$$

Obviously, $I(\theta) = 20 I_1(\theta)$. So we have known $I(\theta) = 10$. Then,

$$\theta^{(t+1)} = \theta^{(t)} + I\left(\theta^{(t)}\right)^{-1} l'\left(\theta^{(t)}\right) = \theta^{(t)} + \frac{l'\left(\theta^{(t)}\right)}{10}$$

We write the R code based on this formula with the 10 initial values of the previous question for calculation.

```
x = c(1.77, -0.23, 2.76, 3.80, 3.47, 56.75, -1.34, 4.24, -2.44,
      3.29, 3.71, -2.40, 4.53, -0.07, -1.05, -13.87, -2.53, -1.75,
      0.27, 43.21)
xm=mean(x)
lp = function(t){
  return(sum((x-t)/(1+(x-t)^2)))
}
xin = c(-11, -1, 0, 1.5, 4, 4.7, 7, 8, 38)
FS = function(x0 = 1, N = 20, tol = 1e-12){ i=1
p = x0
s = lp(x0)/10
while(i <= N & abs(s) > tol){
  p = c(p, p[i]+s)
  i = i+1
  s = lp(p[i])/10
}
return(list(root = p[i], process = p[2:i], nit = i-1))
}
xin = c(xin, xm)
sapply(xin, function(y){FS(x0 = y, N = 1000)})
```

We get the result

X_0	**-11**	**-1**	**0**	**1.5**	**4**	**4.7**	**7**	**8**	**38**	**mean(x)=5.106**
Root	-0.1922866	-0.1922866	-0.1922866	-0.1922866	2.817472	2.817472	2.817472	2.817472	2.817472	2.817472
Nit	202	155	144	180	210	212	218	221	686	213

In the Newton Raphson algorithm, the second derivative (matrix) of the loss function needs to be obtained during parameter estimation. In Fisher scoring, we use the expectation of the second derivative matrix instead. This is the difference between the two. In GLM, when the link function is canonical link, the two algorithms are equivalent, such as logistic regression.

In conclusion, Fisher scoring is a hill-climbing algorithm for getting results - it maximizes the likelihood by getting successively closer and closer to the maximum by taking another step (an iteration). It knows when it has reached the top of the hill in that taking another step does not increase the likelihood.

hw9 第一题

证明：

我们先证明一个引理，即如果将对称矩阵的第一列进行高斯消元后，得到的矩阵的去除第 1 行与第 1 列得到的子矩阵仍然是对称矩阵。

我们假设对称矩阵 $A=\begin{bmatrix} a_{11} & w^T \\ w & A^* \end{bmatrix}$， 用 a_{11} 消去第一列中其他元素，得到 $A'=\begin{bmatrix} a_{11} & w^T \\ 0 & A^*-a_{11}^{-1}w^Tw \end{bmatrix}$，已知 A^* 为对称矩阵，从而$A^*-a_{11}^{-1}w^Tw$ 也为对称矩阵，引理得证。

回到原题, 我们不妨记对矩阵 A 高斯消元后得到的矩阵为 A' 。同时，我们取

$$\{v_0,v_1,v_2,\ldots,v_{n-1}\}=\{e_1,e_2,e_3\ldots,e_n\}$$

为 n 个线性无关的 n 维列向量。利用Gram-Schmidt 方法生成一组 A-共轭的 n 维列向量 $\{d_0,d_1,d_2,\ldots,d_{n-1}\}$。接下来我们记

$$L=[d_0\,|d_1|\,d_2|\ldots|d_{n-1}]^T$$

我们提出命题 $A'=LA$, 也就是说

$$A'=[Ad_0\,|Ad_1|\,Ad_2|\ldots|Ad_{n-1}]^T$$

如果证明了上述命题，我们取 $x=\alpha_0d_0+\alpha_1d_1+\alpha_2d_2+\cdots+\alpha_{n-1}d_{n-1}$，接下来， 求解 $LAx=Lb$ 。 由于 d_i 是 A —共轭的向量, 上述方程等价于下面的 n 个方程：

$$\alpha_i*{d_i}^TAd_i={d_i}^Tb,i=0,1,2\ldots n-1$$

由此解出 $\alpha_i=\frac{{d_i}^Tb}{{d_i}^TAd_i}$。

事实上我们就可以将高斯消元法视作依次求解上述的 n 个方程的过程，最终得到 $x=\sum_{i=0}^{n-1}\frac{{d_i}^Tb}{d_i^TAd_i}d_i$。（这一过程可以视为进行 n 次迭代， 即 $x_{k+1}=x_k+\alpha_kd_k,k=0,1,2..n-1,x_0=0$） 再考虑将 $\{d_0,d_1,d_2,\ldots,d_{n-1}\}$ 作为 A —共轭的 n 维向量时，利用共轭方向法求解 x 的过程中, 迭代式为 $x_{k+1}=x_k+\alpha_kd_k$，其中

$$\alpha_k=-\frac{d_k^T\nabla f(x_k)}{{d_k}^TAd_k}=-\frac{{d_k}^TAx_k}{{d_k}^TAd_k}+\frac{{d_k}^Tb}{{d_k}^TAd_k}$$

事实上, 考虑到 x_k 是 $\{d_0,d_1,d_2,\ldots,d_{k-1}\}$ 的线性组合, 从而上式第一项为 0 , 从而 $\alpha_k=\frac{{d_k}^Tb}{{d_k}^TAd_k}$ 。这也就与高斯消元法中的结果完全相同, 由此我们就说明了高斯消元法是共轭方向法的一种的特殊情况，也就是说, 如果选取特定的初始线性无关向量, 那么共轭方向法就与高斯消元法相同。

最后我们完成之前提出的命题的证明, 即证明 $A'=LA$ 。 我们利用归纳法对A' 的每一行依次进行证明。 易知 A' 的第一行满足条件, 即 ${e_1}^TA'={e_1}^TA={d_0}^TA$ 我们假设对前 $i-1$ 行命题成立, 即 ${e_j}^TA'={d_{j-1}}^TA,j=1,2,\ldots,i-1$，对 A' 的第 i 行进行讨论。下面的叙述中，我们取 $j=1,2,\ldots,i-1$, 考虑完成了前 $j-1$ 列的消去后, 得到的矩阵为 $A^{(j-1)}=\begin{bmatrix} A'_{j-1} & w \\ 0 & B_{n-j} \end{bmatrix}$。 由引理知 B_{n-j} 为对 称矩阵，从而我们对第 j 列进行消元时，考虑到 B_{n-j} 的对称性，第 i 行的计算式为：

$$e_i^TA^{(j)}=e_i^TA^{(j-1)}-e_j^TA^{(j-1)}*\frac{{e_i}^TA^{(j-1)}e_j}{{e_j}^TA^{(j-1)}e_j}={e_i}^TA^{(j-1)}-{e_j}^TA^{(j-1)}*\frac{{e_j}^TA^{(j-1)}e_i}{{e_j}^TA^{(j-1)}e_j}$$

考虑到由归纳假设可知 $e_j^TA^{(j-1)}={d_{j-1}}^TA$， 从而

$$e_i{}^T A^{(j)} = e_i{}^T A^{(j-1)} - d_{j-1}{}^T A * \frac{d_{j-1}{}^T A e_i}{d_{j-1}{}^T A e_j} = e_i{}^T A^{(j-1)} - d_{j-1}{}^T A * \frac{d_{j-1}{}^T A v_{i-1}}{d_{j-1}{}^T A d_{j-1}}$$

上述第二个等号是由于 $d_{j-1} = e_j + l\left(d_0, d_1, d_2, \ldots, d_{j-2}\right)$, 其中 l（ ） 表示线性组合。又因为 d_i 是 A —共轭的, 从而 $d_{j-1}{}^T A d_{j-1} = d_{j-1}{}^T A e_j$ 。我们对 $j = 1, 2, \ldots, i$ 重复上述过程， 最终得到

$$e_i^T A' = e_i^T A - \sum_{j=1}^{i-1} d_{j-1}^T \frac{d_{j-1}^T A v_{i-1}}{d_{j-1}{}^T A d_{j-1}} * A$$

另一方面, 由生成 d_i 的 Gram-Schmidt 方法可知,

$$d_{i-1} = v_{i-1} - \sum_{j=0}^{i-2} \frac{d_j^T A v_{i-1}}{d_j^T A d_j} d_j = v_{i-1} - \sum_{j=1}^{i-1} \frac{d_{j-1}^T A v_{i-1}}{d_{j-1}^T A d_{j-1}} d_{j-1}$$

结合上面两个式子可知 $e_i{}^T A' = d_{i-1}{}^T A$, 从而归纳法成立, 即这个式子对$i = 1, 2, \ldots, n$ 都成立。结合 n 个式子即得

$$A' = \left[Ad_0 \left|Ad_1\right| Ad_2 \right| \ldots \left|Ad_{n-1}\right]^T = LA$$

这样我们就完成了“ $A' = LA$ “这一命题的证明，结含之前的叙述，我们也就完成了“高斯消元法是共轩方向法的特例”这一命题的证明。

hw11 第一题

证明:

先设 $X_{i+1} \equiv aX_i (\bmod m)$ 对任意 i 成立，即存在整数 d 使得 $X_{i+1} - aX_i = md_i$, 故

$$X_{i+1} = aX_i + md_i = a\left(aX_{i-1} + md_{i-1}\right) + md_i = \cdots = a^{i+1} X_0 + \left(d_i + ad_{i-1} + \cdots + a^i d_0\right) m$$

从而 $X_{i+1} \equiv a^{i+1} X_0 (\bmod m)$ 对任意 i 成立。

反之, 设 $X_{i+1} \equiv a^{i+1} X_0 (\bmod m)$ 对任意 i 成立，则存在整数 d_i 使得 $X_i - a^i X_0 = md_i$, 从 而

$$X_{i+1} = a^{i+1} X_0 + md_{i+1} = a\left(X_i - md_i\right) + md_{i+1} = aX_i + m\left(d_{i+1} - ad_i\right)$$

因此 $X_{i+1} \equiv aX_i (\bmod m)$ 对任意 i 成立。