Exercise 3:

(a) Choose the best one according to minimum AIC:
R-code:

```
#fitting the models:
model.GammaLog <- glm(Claim ~ CarType + DriverAge, data = ClaimData, family = Gamma("log"))
model.NormalLog <- glm(Claim ~ CarType + DriverAge, data = ClaimData, family = gaussian("log"))
model.InverseGaussianLog <- glm(Claim ~ CarType + DriverAge, data = ClaimData, family = inverse.gaussian("log"))

#compare AIC of the models:
AIC(model.GammaLog)
AIC(model.NormalLog)
AIC(model.InverseGaussianLog)
```

R-output:

```
> #fitting the models:
> model.GammaLog <- glm(Claim ~ CarType + DriverAge, data = ClaimData, family = Gamma("log"))
> model.NormalLog <- glm(Claim ~ CarType + DriverAge, data = ClaimData, family = gaussian("log"))
> model.InverseGaussianLog <- glm(Claim ~ CarType + DriverAge, data = ClaimData, family = inverse.gaussian("log"))
>
> #compare AIC of the models
> AIC(model.GammaLog)
[1] 151.2867
> AIC(model.NormalLog)
[1] 152.8516
> AIC(model.InverseGaussianLog)
[1] 150.3136
```

Comment: AIC is a popular criterion for model selection, which balances goodness-of-fit (expressed as the achieved log-likelihood) and number of parameters. The smaller AIC, the better. All these models have same number of parameters and so the comparison by AIC is equivalent to comparison of deviance statistics (D). Based on the comparison of AIC, we choose the third model with Inverse Gaussian error distribution and log-link.

(b) For the chosen model, assess the possibility to reduce variables by performing backward stepwise variable selection (use AIC criterion).
R-code:

```
#backwards stepwise selection based on minimiztion of AIC for the chosen model
backwards <- step(model.InverseGaussianLog)
```

R-output:

```
> backwards <- step(model.InverseGaussianLog)
Start:  AIC=150.31
Claim ~ CarType + DriverAge

            Df   Deviance    AIC
<none>         1.1294e-05 150.31
- CarType    2 4.5581e-05 164.91
- DriverAge  3 1.7421e-04 232.66
```

Comment: If we drop CarType variable from the model, AIC would increase to 164.91, which is undesirable. Droping the variable DriverAge would increase AIC even more (up to 232.66). Hence, backward stepwise procedure did not drop any variable from the model and kept it unchanged.

(c) Review the results of variable reduction analysis from (b) by performing F test for sub-models. R-code:

```
# test if we can drop CarType by assessing the model differences with F test:
model.reduced <- update(model.InverseGaussianLog, . ~ . - CarType)
anova(model.reduced, model.InverseGaussianLog, test = "F")


# test if we can drop DriverAge by assessing the model differences with F test:
model.reduced <- update(model.InverseGaussianLog, . ~ . - DriverAge)
anova(model.reduced, model.InverseGaussianLog, test = "F")
```

R-output:

```
> # test if we can drop CarType by assessing the model differences with F test
> model.reduced <- update(model.InverseGaussianLog, . ~ . - CarType)
> anova(model.reduced, model.InverseGaussianLog, test = "F")
Analysis of Deviance Table

Model 1: Claim ~ DriverAge
Model 2: Claim ~ CarType + DriverAge
  Resid. Df Resid. Dev Df   Deviance      F  Pr(>F)
1         8 4.5581e-05
2         6 1.1294e-05  2 3.4287e-05 9.2968 0.01452 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
>
> # test if we can drop DriverAge by assessing the model differences with F test
> model.reduced <- update(model.InverseGaussianLog, . ~ . - DriverAge)
> anova(model.reduced, model.InverseGaussianLog, test = "F")
Analysis of Deviance Table

Model 1: Claim ~ CarType
Model 2: Claim ~ CarType + DriverAge
  Resid. Df Resid. Dev Df   Deviance     F   Pr(>F)
1         9 1.7421e-04
2         6 1.1294e-05  3 0.00016291 29.45 0.000549 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

Comment: p-value of the F-test for dropping CarType is 0.01452 and is significant on 5 % level (but not on 1 % level). This means that probability of type 1 error (rejecting valid H_0) is 0.01452. In other words, if we do NOT drop CarType (reject H_0), we are wrong with probability 0.01452 (given the data and model). This confirms our decision from step (b).
The F-test for dropping DriverAge has p-value as small as 0.000549. Hence, if we dropped DriverAge from the model, the increase in Deviance would be even more significant and so this variable is more important for the model and we keep it there. This is again confirms our conclusions from step (b).