Modellazione statistica e grafica su GSS

Riccardo Fantechi

Università degli Studi di Firenze

20 giugno 2025

Abstract

In questo progetto analizzeremo le opinioni della popolazione statunitense rispetto all'introduzione di leggi restrittive sul possesso di arma da fuoco. Nello specifico studiamo come l'opinione su questa legge, da parte degli individui, sia influenzata dal background di quest'ultimi, come per esempio dal sesso dell'intervistato o alle opinioni di esso riguardo la pena di morto o alla legalizzazione dell'aborto per i casi di violenza.

Contents

1 Introduzione

Il dataset analizzato è il **General Social Survey GSS**, una delle fonti più utilizzate nella ricerca sociale americana, in quanto raccoglie annualmente informazioni dettagliate su opinioni, comportamenti e caratteristiche personali di un ampio campione rappresentativo della popolazione statunitense.
In questa analisi, oltre alla variabile target *GUNLAW*, verranno considerate alcune variabili di background che si ritiene possano influenzare l'opinione degli intervistati:

- **SEX**, sesso dell'intervistato.
- **SATJOB**, il grado di soddisfazione sul posto di lavoro.
- **CONFINAN**, il livello di fiducia nelle istituzioni finanziarie.
- **CAPPUN**, opinione sulla pena di morte.
- **ABRAPE**, opinione sull'aborto in caso di stupro.

Scopo ultimo dell'elaborato è identificare, tramite diverse tecniche di modellazione statistica, quali fattori influenzino maggiormente il supporto alla legge sulle armi da fuoco. Verranno pertanto confrontati modelli diversi, anche tramite criteri di selezione, al fine di individuare la struttura più informativa tra le variabili considerate.

1.1 Software

La modellazione statistica dei dati è stata eseguita tramite il linguaggio **R**, organizzando il dataset e le cartelle all'interno dell'ambiente **RStudio**. Sono state utilizzate diverse librerie per la visualizzazione grafica, per la costruzione di grafi indiretti, ovvero la *log-lineare*, per le reti *Bayesiane* e per la gestione dei valori mancanti.
L'intero progetto è documentato su Latex, includendo immagini e script di codici su cui porre l'attenzione.

https://github.com/Riccardo1451/FSM_GunLaw

---

2 Preprocessing del dataset

2.1 Analisi e visualizzazione del dataset

Prima di iniziare a lavorare sulla statistica, è necessario visualizzare e analizzare il dataset in soggetto. Vogliamo avere un'idea di cosa contenga e sotto quale formato. Importante è anche la parte di pulizia e di purificazione dei dati.

```
>#  Caricamento del dataset
>load("GSS.RData")
>#  Visualizzazione delle variabili contenute nel DS
>#  insieme al sommario statistico
>summary(GSS)

     CAPPUN          GUNLAW           SEX           ABRAPE          CONFINAN
 Min.   :1.00    Min.   :1.00    Min.   :1.00   Min.   :1.000   Min.   :1.00
 1st Qu.:1.00    1st Qu.:1.00    1st Qu.:1.00   1st Qu.:1.000   1st Qu.:1.00
 Median :1.00    Median :1.00    Median :2.00   Median :1.000   Median :2.00
 Mean   :1.27    Mean   :1.23    Mean   :1.56   Mean   :1.178   Mean   :1.89
 3rd Qu.:2.00    3rd Qu.:1.00    3rd Qu.:2.00   3rd Qu.:1.000   3rd Qu.:2.00
 Max.   :2.00    Max.   :2.00    Max.   :2.00   Max.   :2.000   Max.   :3.00
 NA's   :10384   NA's   :17986                  NA's   :15784   NA's   :20006
     SATJOB
 Min.   :1.000
 1st Qu.:1.000
 Median :2.000
 Mean   :1.667
 3rd Qu.:2.000
 Max.   :3.000
 NA's   :14264

>#Visualizzazione dela struttura del DS
>str(GSS)

'data.frame': 51020 obs. of  6 variables:
 $ CAPPUN  : int  1 1 2 2 2 1 2 NA 1 2 ...
 $ GUNLAW  : int  2 NA NA 1 1 NA 1 NA 1 1 ...
 $ SEX     : int  2 1 2 2 1 2 2 2 2 2 ...
 $ ABRAPE  : int  2 NA NA 1 1 NA 1 NA 1 1 ...
 $ CONFINAN: int  1 1 2 NA NA NA 3 NA 1 NA ...
 $ SATJOB  : num  1 1 NA 2 NA 1 1 NA 3 1 ...
```

Da questa prima analisi notiamo che il dataset contiene 6 variabili tutte discrete, quattro sono di tipo binario e due ordinali, nel dettaglio:

- CAPPUN: 1 -> Favorevole, 2 -> Contrario.
- GUNLAW: 1 -> Favorevole, 2 -> Contrario.
- SEX: 1 -> Maschio, 2 -> Femmina.
- ABRAPE: 1 -> Favorevole, 2 -> Contrario.
- CONFINAN: 1 -> Fiducioso, 2 -> Neutrale, 3 -> Scettico.
- SATJOB: 1 -> Soddisfatto, 2 -> Neutrale, 3 -> Insoddisfatto.

2.2 Rimozione dei valori mancanti

La prima cosa che salta all'occhio è la presenza di molti valori mancanti segnati con la sigla **NA** (Not Available). Si possono adottare diversi approcci per la risoluzione di questo problema tra cui, rimuovere completamente le righe manchanti, sostituire i valori mancanti con una stima (media, moda) o utilizzare metodi statistici che funzionano anche con valori NA. In questa analisi per semplicità eliminiamo le righe che contengono valori non validi. Riteniamo importante notare che il numero di osservazioni cala drasticamente dopo questo passaggio, andando da 51020 a 13067, sottolineando la complessità nel raccogliere dei dati utilizzabili e il mantenimento di essi.

```
> #Eliminazione dei valori mancanti/non validi
> GSS_cleaned <- na.omit(GSS)
> #Visualizzazione del DS pulito
> str(GSS_cleaned)


'data.frame': 13067 obs. of  6 variables:
 $ CAPPUN  : int  1 2 1 2 2 2 1 2 1 1 ...
 $ GUNLAW  : int  2 1 1 1 1 1 1 1 1 1 ...
 $ SEX     : int  2 2 2 2 2 2 1 2 1 2 ...
 $ ABRAPE  : int  2 1 1 1 1 1 1 2 1 1 ...
 $ CONFINAN: int  1 3 1 2 1 2 2 1 2 2 ...
 $ SATJOB  : num  1 1 3 1 1 1 2 1 1 1 ...
```

2.3 Codifica delle variabili in fattori

In R, le variabili di tipo *factor* sono utili quando si lavora con dati qualitativi, poiché permettono di trarre le modalità come categorie distinte, e non come valori numerici interi.
La trasformazione è stata fatta utilizzando la funzione *factor()*, che permette di compiere l'azione voluta con l'aggiunta di un etichetta alla categoria, in modo da semplificarne la lettura.

```
> #Trasformazione delle variabili in fattori
> col_to_factors <- c("CAPPUN", "GUNLAW", "SEX", "ABRAPE", "CONFINAN","SATJOB")
> GSS_cleaned[col_to_factors] <- lapply(GSS_cleaned[col_to_factors], factor)
>
> #Aggiunta di etichette per migliorare la leggibilit
>
> #CAPPUN
> GSS_cleaned$CAPPUN <- factor(GSS_cleaned$CAPPUN,
+                              levels = c(1, 2),
+                              labels = c("Favorevole", "Contrario"))
>
> #GUNLAW
> GSS_cleaned$GUNLAW <- factor(GSS_cleaned$GUNLAW,
+                              levels = c(1, 2),
+                              labels = c("Favorevole", "Contrario"))
>
> #SEX
> GSS_cleaned$SEX <- factor(GSS_cleaned$SEX,
```

```
+                             levels = c(1, 2),
+                             labels = c("Maschio", "Femmina"))
>
> #ABRAPE
> GSS_cleaned$ABRAPE <- factor(GSS_cleaned$ABRAPE,
+                               levels = c(1, 2),
+                               labels = c("Favorevole", "Contrario"))
>
> #CONFINAN: fiducia nel governo
> GSS_cleaned$CONFINAN <- factor(GSS_cleaned$CONFINAN,
+                                 levels = c(1, 2, 3),
+                                 labels =c("Fiducioso",
                                           "Neutrale",
                                           "Scettico"))
>
> #SATJOB
> GSS_cleaned$SATJOB <- factor(GSS_cleaned$SATJOB,
+                               levels = c(1, 2, 3),
+                               labels = c("Soddisfatto",
+                                          "Neutrale",
+                                          "Insoddisfatto"))
> #Visualizzazione del DS dopo la fattorizzazione
> str(GSS_cleaned)

'data.frame': 13067 obs. of  6 variables:
 $ CAPPUN  : Factor w/ 2 levels "Favorevole","Contrario": 1 2 1 2 2 2 1 2 1 1 ...
 $ GUNLAW  : Factor w/ 2 levels "Favorevole","Contrario": 2 1 1 1 1 1 1 1 1 1 ...
 $ SEX     : Factor w/ 2 levels "Maschio","Femmina": 2 2 2 2 2 2 1 2 1 2 ...
 $ ABRAPE  : Factor w/ 2 levels "Favorevole","Contrario": 2 1 1 1 1 1 1 2 1 1 ...
 $ CONFINAN: Factor w/ 3 levels "Fiducioso","Neutrale",..: 1 3 1 2 1 2 2 1 2 2 ...
 $ SATJOB  : Factor w/ 3 levels "Soddisfatto",..: 1 1 3 1 1 1 2 1 1 1 ...
```

In questo modo, l'analisi statistica e la visualizzazione dei dati risultati risulteranno più chiare e coerenti, poiché le modalità sono trattate esplicitamente come categorie. Questo risulta particolarmente importante per i grafici e i modelli statistici che utilizzeremo più avanti.

3 Analisi esplorativa

In questa sezione ci concentreremo sulla variabile *CAPPUN* che rileva se l'individuo è favorevole alla pena di morte per omicidio. Si tratta di una variabile categorica binaria. Analizzeremo la sua distribuzione, le relazioni con altre variabili del dataset, e costruiremo un modello esplicativo. Per poter notare eventuali relazioni di dipendenza tra le variabili abbiamo optato per tre diversi categorie di visualizzazione:

- Frequenze, possiamo visualizzare tramite grafico a barre il numero di individui che la pensano in un modo rispetto all'altro, comodo per avere una visione rapida sul trend.
- Matrici di contingenza, riportano i vari livelli di diverse variabili su righe e colonne.
- Heatmap, è un metodo di visualizzazione dei dati che permette di visualizzare i valori di una tabella grazie ad una colorazione.

3.1 Frequenze

```
> #Visualizzaione grafico a barre
> library(ggplot2)
> ggplot(GSS_cleaned, aes(x = CAPPUN)) +
+   geom_bar(fill = "purple3") +
+   geom_text(stat = "count", aes(label = ..count..), vjust = -0.5) +
+   labs(title = "Distribuzione di CUPPUN",
+        x = "Opinione", y = "Frequenza") +
+   theme_minimal()
```

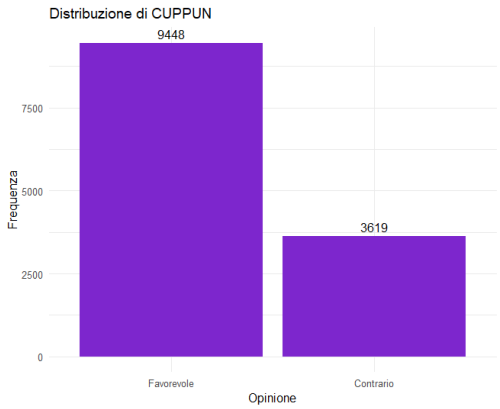

Figura 1: Distribuzione CAPPUN

3.2 Matrici di contingenza

Di seguito troviamo tutte le matrici di contingenza, da cui con pochi semplici calcoli, possiamo ricavare la percentuale della probabilità marginale con cui un evento si può verificare.

```
> #Selezione delle variabili di confronto
> variabili_confronto <- c("SEX", "SATJOB", "ABRAPE", "GUNLAW", "CONFINAN")
> #Ciclo per visualizzazione della matrice di contingenza per ogni variabile
> for (var in variabili_confronto) {
+
+   tab <- table(GSS_cleaned$CAPPUN, GSS_cleaned[[var]])
+   dimnames(tab)[[1]] <- levels(GSS_cleaned$CAPPUN)
+   dimnames(tab)[[2]] <- levels(GSS_cleaned[[var]])
+   names(dimnames(tab)) <- c("CAPPUN", var)
+
+   #Matrice totale
+   cat("\n========================================\n")
+   cat("Matrice di contingenza assoluta: CAPPUN vs", var, "\n")
+   cat("========================================\n")
+   print(tab)
+
+   #Matrice percentuale
+   prop_tab <- round(prop.table(tab, margin = 2) * 100, 1)
+
+   cat("\n----------------------------------------\n")
+   cat("Distribuzione % per colonna: CAPPUN vs", var, "\n")
+   cat("----------------------------------------\n")
+   print(prop_tab)
+ }


========================================
Matrice di contingenza assoluta: CAPPUN vs SEX
========================================
            SEX
CAPPUN       Maschio Femmina
  Favorevole    4432    5016
  Contrario     1297    2322

----------------------------------------
Distribuzione % per colonna: CAPPUN vs SEX
----------------------------------------
            SEX
CAPPUN       Maschio Femmina
  Favorevole    77.4    68.4
  Contrario     22.6    31.6

========================================
```

```
Matrice di contingenza assoluta: CAPPUN vs SATJOB
=======================================
            SATJOB
CAPPUN       Soddisfatto Neutrale Insoddisfatto
  Favorevole        4628     3495          1325
  Contrario         1637     1375           607

---------------------------------------
Distribuzione % per colonna: CAPPUN vs SATJOB
---------------------------------------
            SATJOB
CAPPUN       Soddisfatto Neutrale Insoddisfatto
  Favorevole        73.9     71.8          68.6
  Contrario         26.1     28.2          31.4

=======================================
Matrice di contingenza assoluta: CAPPUN vs ABRAPE
=======================================
            ABRAPE
CAPPUN       Favorevole Contrario
  Favorevole       8070      1378
  Contrario        2790       829

---------------------------------------
Distribuzione % per colonna: CAPPUN vs ABRAPE
---------------------------------------
            ABRAPE
CAPPUN       Favorevole Contrario
  Favorevole       74.3      62.4
  Contrario        25.7      37.6

=======================================
Matrice di contingenza assoluta: CAPPUN vs GUNLAW
=======================================
            GUNLAW
CAPPUN       Favorevole Contrario
  Favorevole       7023      2425
  Contrario        2922       697

---------------------------------------
Distribuzione % per colonna: CAPPUN vs GUNLAW
---------------------------------------
            GUNLAW
CAPPUN       Favorevole Contrario
  Favorevole       70.6      77.7
```

```
  Contrario        29.4      22.3

=======================================
Matrice di contingenza assoluta: CAPPUN vs CONFINAN
=======================================
            CONFINAN
CAPPUN       Fiducioso Neutrale Scettico
  Favorevole      2760     5298     1390
  Contrario       1035     2003      581

---------------------------------------
Distribuzione % per colonna: CAPPUN vs CONFINAN
---------------------------------------
            CONFINAN
CAPPUN       Fiducioso Neutrale Scettico
  Favorevole      72.7     72.6     70.5
  Contrario       27.3     27.4     29.5
```

Riassumendo alcuni di questi risultati notiamo:

- SEX: il **77.4**% dei maschi sia favorevole alla pena di morte, rispetto alla percentuale delle femmine del **68.4**%. Data la differenza significativa tra le due percentuali possiamo intuire che esiste una dipendenza tra le due variabili, concludendo che è più probabile che un maschio sia favorevole alla legge sulla pena di morte rispetto ad una femmina. Il calcolo che viene svolto, lo si può vedere anche dal codice, è una probabilità condizionata di *CAPPUN* dato il livello di *SEX* = *Maschio/Femmina*.
- SATJOB: notiamo come non ci sia molta variazione di pensiero tra le varie categorie, ad eccezione di chi è soddisfatto del lavoro che tende ad essere favorevole alla pena di morte, rispetto a chi è insoddisfatto del lavoro che è meno propenso alla pena di morte. Questo ci suggerisce che le due variabili non siano strettamente correlate.
- ABRAPE: Notiamo che le persone a favore dell'aborto per le vittime di violenze e a favore della pena di morte siano maggiori delle persone contro la legge sull'aborto e a favore delle pena di morte, il che ci indica una possibile relazione tra le variabili.
- GUNLAW: la percentuale di individui favorevoli a una legge sul porto d'armi e alla pena di morte sia minore dei contrari alla legge sulle armi e favorevole alla pena. Da questo, si può intuire una possibile relazione tra le due.
- CONFINAN:ci mostra come non ci sia una grande variazione percentuale tra la fiducia nelle istituzione e la pena di morte.

3.3 Heatmap

In questa sezione abbiamo riportato gli stessi dati sotto forma di heatmap, ovvero una rappresentazione visiva dei dati, dove i singoli valori vengono mostrati tramite colori, dove il colore più acceso indica un valore maggiore, viceversa per un colore più chiaro:

```
> #Visualizzazione heatmap
> for (var in variabili_confronto) {
+
+   # Tabella di contingenza
+   tab <- table(GSS_cleaned$CAPPUN, GSS_cleaned[[var]])
+
+   # Calcolo percentuali colonna per colonna
+   tab_perc <- round(prop.table(tab, margin = 2) * 100, 1)
+
+   # Conversione
+   df_heat <- as.data.frame(tab_perc)
+   names(df_heat) <- c("CAPPUN", "Variabile", "Percentuale")
+   df_heat$CAPPUN <- factor(df_heat$CAPPUN, levels = c("Contrario",
                                                       "Favorevole"))
+
+   # Heatmap con percentuali
+   print(
+   ggplot(df_heat, aes(x = Variabile, y = CAPPUN, fill = Percentuale)) +
+     geom_tile(color = "white") +
+     geom_text(aes(label = paste0(Percentuale, "%")), color = "black",
                                                        size = 4)
+     scale_fill_gradient(low = "white", high = "firebrick") +
+     labs(title = paste("Heatmap␣percentuale:␣CAPPUN␣vs", var),
+          x = var, y = "CAPPUN") +
+     theme_minimal()
+   )
+ }
```

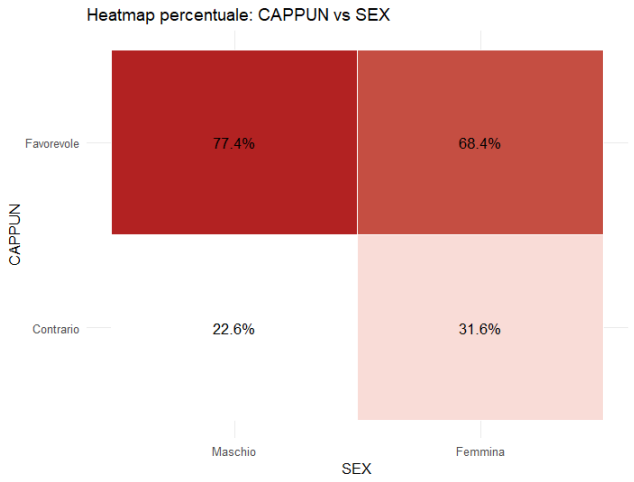

Figura 2: Heatmap CAPPUN vs SEX

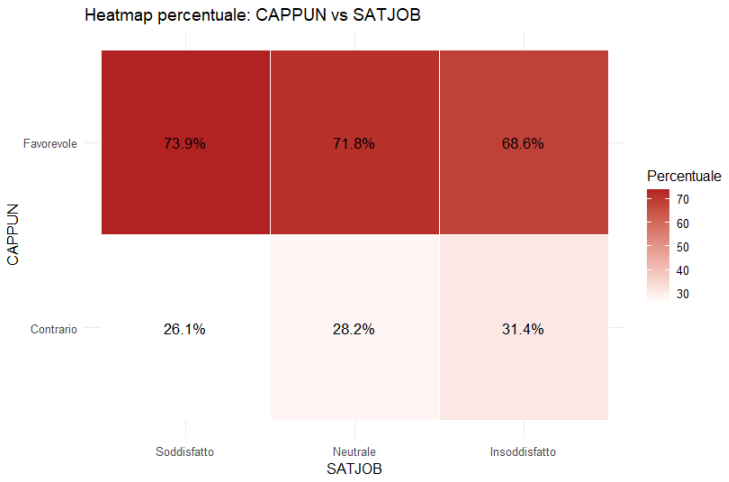

Figura 3: Heatmap CAPPUN vs SATJOB

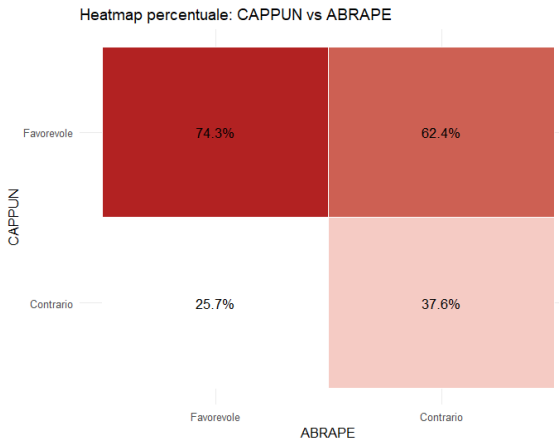

Figura 4: Heatmap CAPPUN vs ABRAPE

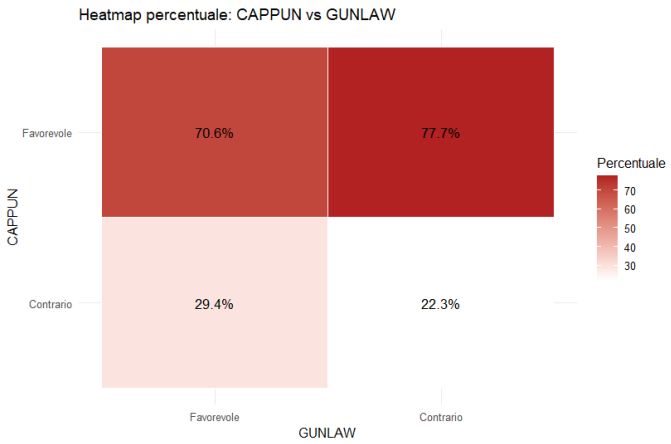

Figura 5: Heatmap CAPPUN vs GUNLAW

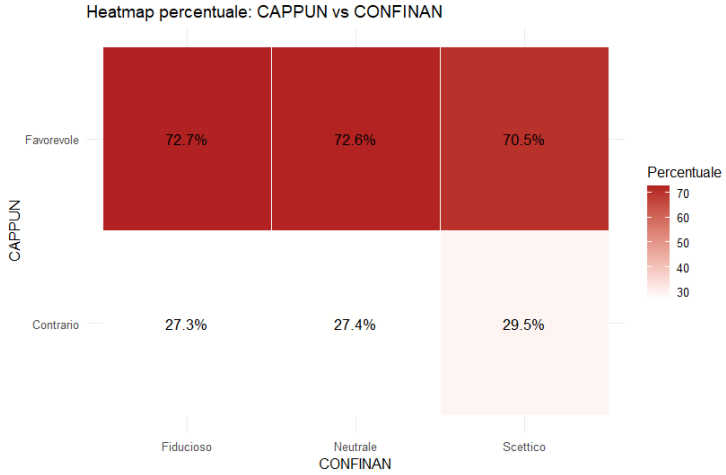

Figura 6: Heatmap CAPPUN vs CONFINAN

4 Modelli grafici indiretti

I modelli grafici indiretti sono una classe di modelli statistici utilizzati per rappresentare le dipendenze tra variabile categoriche, come nel nostro caso, attraverso dei grafi non diretti. Stiamo trattando un estensione dei modelli *log-lineari* per permettere una visualizzazione intuitiva delle relazioni tramite le variabili (nodi) e gli archi (interazioni).
Di seguito troviamo i pacchetti utilizzati per la costruzione dei grafi, insieme al codice per la visualizzazione.

```
> library(gRim)
> library(gRain)
> library(gRbase)
> #Creazione del modello saturo
> model_sat <- dmod(~.^., GSS_cleaned)
>
> #Creazione del modello di indipendenza
> model_ind <- dmod(~.^1, GSS_cleaned)
>
> #Grafo backward con penalizzazione AIC (Default)
> backAIC_model <- stepwise(model_sat)
> plot(backAIC_model)
> title(main="UG␣backward␣AIC")
>
> #Grafo backward con penalizzazione BIC
> backBIC_model <- stepwise(model_sat, k = log(nrow(GSS_cleaned)))
> plot(backBIC_model)
> title(main="UG␣backward␣BIC")
>
> #Grafo forward con penalizzazione AIC
> forwardAIC_model <- stepwise(model_ind, direction="forward")
> plot(forwardAIC_model)
> title(main="UG␣forward␣AIC")
>
> #Grafo forward con penalizzazione BIC
> forwardBIC_model <- stepwise(model_ind, k = log(nrow(GSS_cleaned)),
+                                              direction="forward")
> plot(forwardBIC_model)
> title(main="UG␣forward␣BIC")
```

È importante soffermarsi su alcune tecniche e funzioni utilizzate per la costruzione di questi grafi. La funzione *dmod()*, del pacchetto *gRim*, consente di definire i modelli log-lineari specificando le interazioni tramite formule, in particolare:

- Il **modello di indipendenza** (modello nullo), indicato con `~.^1`, include solo gli effetti principali.
- Il **modello saturo**, indicato con `~.^.`, considera tutte le possibili interazioni tra le variabili.

Per identificare il modello più adeguato si utilizza la funzione di selezione *stepwise()*. Questa procedura esplora in modo iterativo lo spazio dei modelli log-lineari, che abbiamo definito inizialmente, aggiungendo (*forward*) o rimuovendo (*backward*) interazioni.

4.1 Selezione del modello

La selezione del modello ottimale può essere effettuata mediante criteri di penalizzazione come l'**AIC** (Akaike Information Criterion) e il **BIC** (Bayesian Information Criterion). Questi due criteri bilanciano la bontà di adattamento del modello ai dati forniti, penalizzando il numero di parametri per evitare overfitting.
Il criterio AIC è definito come:

$$AIC = -2 \cdot \log(L) + 2k \tag{1}$$

Mentre il BIC è dato da:

$$BIC = -2 \cdot \log(L) + k \log(n) \tag{2}$$

dove:

- L è la massima verosimiglianza del modello.
- k è il numero di parametri stimati.
- n è la numerosità campionaria.

Nella funzione *stepwise()* abbiamo utilizzato il parametro k che permette di specificare il criterio di penalizzazione.

- **AIC** è il valore di default, ovvero $k = 2$.
- **BIC** si ottiene impostando $k = log(n)$.

Di seguito vediamo i grafici ottenuti:

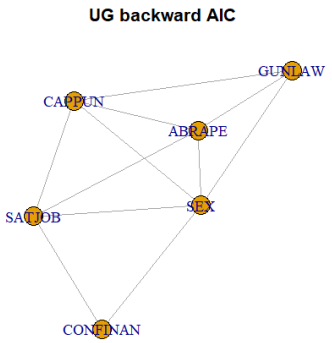

Figura 7: UG backward AIC

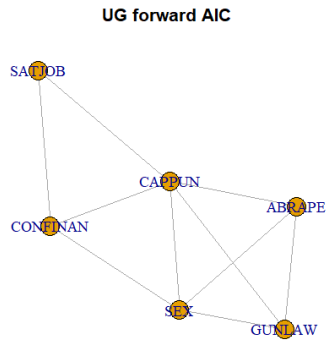

Figura 9: UG forward AIC

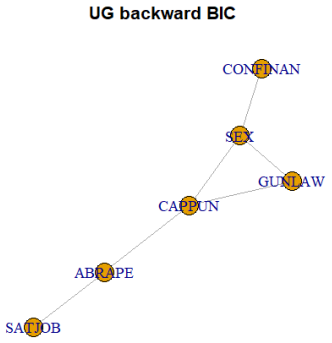

Figura 8: UG backward BIC

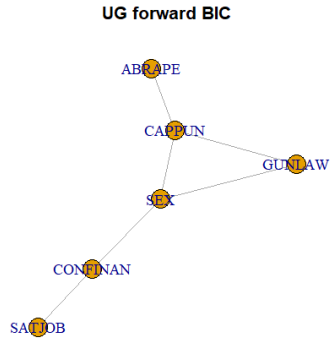

Figura 10: UG forward BIC

Un ultimo metodo molto importante per la selezione dei modelli grafici è il metodo *direction="both"*, per far si che la procedura consideri sia aggiungere che rimuovere interazioni ad ogni passo, partendo da uno dei due modelli iniziali. Questa tecnica è molto utile quando non siamo sicuri se partire dal modello indipendente o da quello saturo e se vogliamo eseguire una ricerca più flessibile, che combini entrambi i vantaggi.

```
> #Grafo both con penalizzazione AIC
> bothAIC_model <- stepwise(model_ind, direction="both")
> plot(bothAIC_model)
> title(main = "UG␣both␣AIC")
>
> #Grafo both con penalizzazione BIC
> bothBIC_model <- stepwise(model_ind, k = log(nrow(GSS_cleaned)),
+                                          direction = "both")
```

```
> plot(bothBIC_model)
> title(main = "UG both BIC")
```

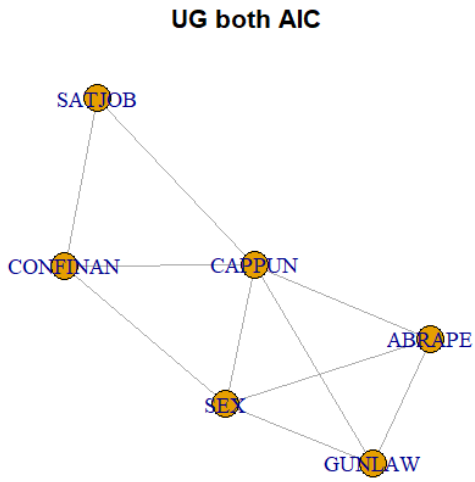

Figura 11: UG both AIC

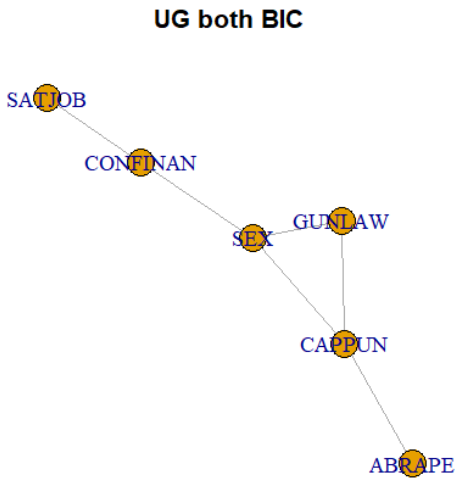

Figura 12: UG both BIC

Si può notare come i modelli trovati tramite BIC siano più parsimoniosi rispetto a quelli con penalizzazione AIC, questo accade perché i modelli AIC premiano la complessità del sistema analizzato favorendo un adattamento più significativo ai dati. Visto che l'obiettivo è quello di trovare un modello che rappresenta meglio il processo con cui sono stati generati i dati si opta sicuramente per un modello con penalizzazione BIC. Per quanto riguarda la direzione del grafo, cerchiamo di sceglierne uno che contenga archi significativi (archi che appaiono anche in altri grafi) rimanendo parsimoniosi. In questo caso, grazie anche all'analisi esplorativa, la decisione cade sul modello BIC forward o il modello BIC both (sono equivalenti).
Anche se la stima dei modelli grafici indiretti è condotta sull'intero dataset, è possibile focalizzare l'attenzione sul nostro caso di studio, ovvero quello della variabile *CAPPUN*. A tale proposito, una volta scelto il modello grafico indiretto, possiamo notare come ci sia una relazione tra la variabile target *CAPPUN* con le variabili *ABRAPE*, *SEX* e *GUNLAW*, come avevamo previsto inizialmente.

5 Modelli grafici diretti

A differenza dei modelli grafici indiretti, i **modelli grafici diretti** rappresentano le relazioni tra le variabili tramite **archi orientati**, dando origine a strutture chiamate **DAG** (Direct Acyclic Graphs). In un DAG ogni nodo rappresenta una variabile, e ogni arco diretto da una freccia, rappresenta una **dipendenza condizionale direzionata**. I modelli diretti sono particolarmente utili quando si vuole modellare la causalità, costruire modello probabilistici interpretabili, o eseguire inferenze condizionali su variabili parzialmente osservabili.

5.1 Reti Bayesiane

Una rete Bayesiana è un modello grafico basato su un DAG che rappresenta congiuntamente:

- Una struttura grafica (le dipendenze tra le variabili).
- Una componente parametrica (le distribuzioni di probabilità condizionate tra le variabili connesse).

Formalmente una rete Bayesiana definisce la distribuzione congiunta come:

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i | Pa(X_i)) \tag{3}$$

dove $Pa(X_i)$ indica i genitori del nodo X_i nel DAG.
Nel nostro caso di studio la struttura della rete è parzialmente sconosciuta perciò deve essere appresa dai dati, tramite la funzione *hc()* che implementa l'algoritmo hill climb. L'algoritmo di tipo scored-based utilizzato per l'apprendimento della struttura, parte da una struttura iniziale e itera migliorando la rete passo dopo passo, cercando di massimizzare un criterio di bontà del modello, come AIC, BIC o log-likelihood. Ad ogni passo hill climb valuta piccole modifiche alla struttura, accettando quella che migliora maggiormente lo score, fino a raggiungere un ottimo locale.

```
> library(bnlearn)
> library(igraph)
>#Creazione del modello Bayesiano
> model_bnstd <- hc(GSS_cleaned)
> dag_bnstd <-as.igraph(model_bnstd)
> plot(
+   dag_bnstd,
+   vertex.size = 20,
+   vertex.label.cex = 0.7,
+   edge.arrow.size = 0.5,
+   main = "Rete␣Bayesiana"
+   )
```

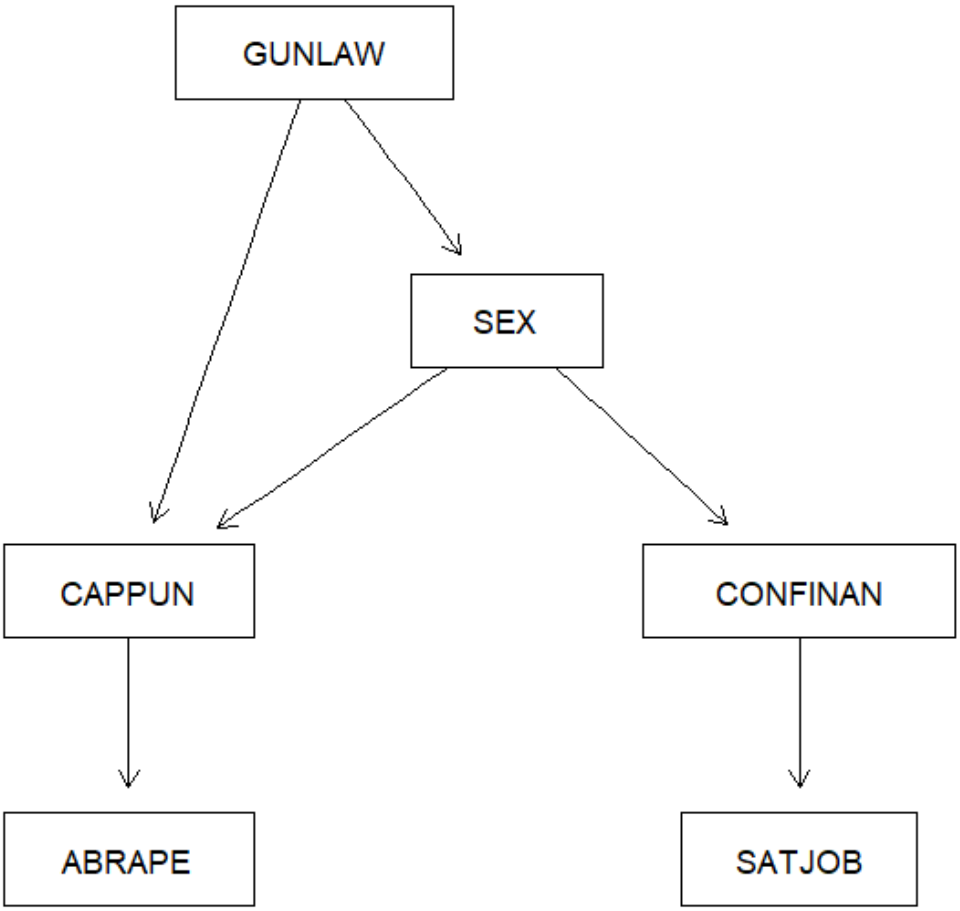

Figura 13: Rete Bayesiana versione classica

5.1.1 Moralizzazione e D-separation

Per approfondire meglio la struttura della dipendenza tra le variabili, abbiamo costruito due modelli di rete Bayesiana utilizzando l'algoritmo *hc()* con due diversi criteri di scoring, AIC e BIC.

```
>#Costruizione dei modelli
>model_bnstd_aic <- hc(GSS_cleaned, score="aic")
>model_bnstd_bic <- hc(GSS_cleaned, score="bic")
>#Visualizzazione dei modelli
>graphviz.plot(model_bnstd_aic)
>graphviz.plot(model_bnstd_bic)
```

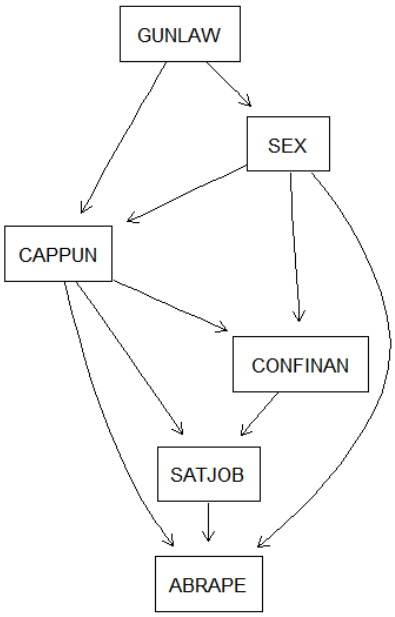

Figura 14: Rete Bayesiana AIC

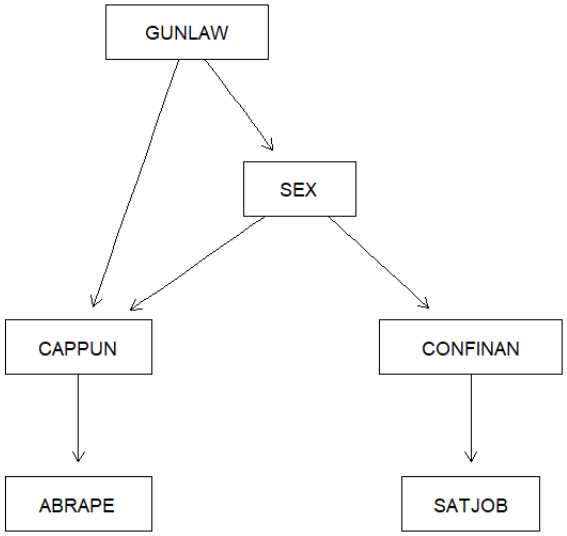

Figura 15: Rete Bayesiana BIC

Successivamente, abbiamo confrontato le due reti ottenute, analizzando se risultassero **markovianamente equivalenti**, per verificarsi:

1. Hanno gli stessi archi non direzionati (stesse adiacenze).
2. Hanno gli stessi v-structures, del tipo A -> C <- B, con A e B non collegati.

Prima di fare ciò abbiamo dovuto moralizzare i due DAG, la moralizzazione è una trasformazione che converte un DAG in un grafo non diretto, per poter confrontare in modo più semplice le indipendenze, in particolare; rende non direzionati tutti gli archi e aggiunge archi tra i genitori comuni.

```
>moral(model_bnstd_aic)
>moral(model_bnstd_bic)
```

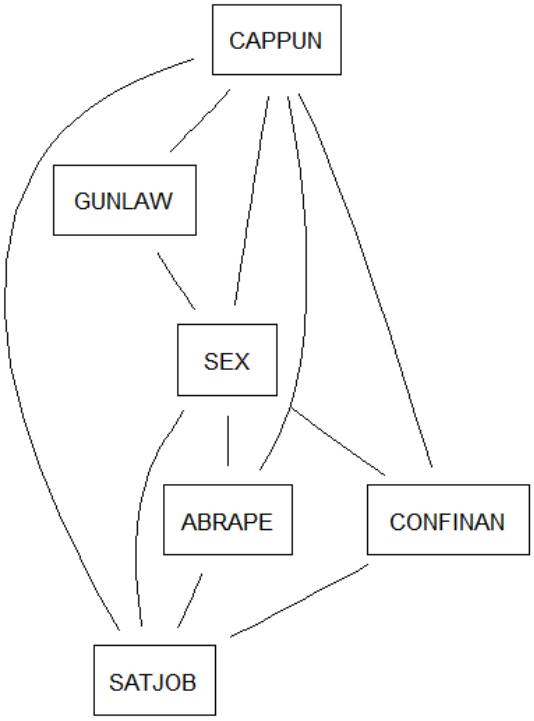

Figura 16: Rete Bayesiana moralizzata AIC

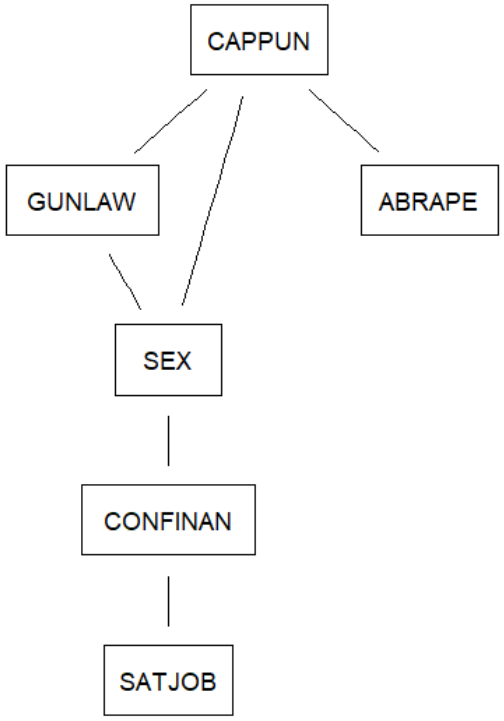

Figura 17: Rete Bayesiana moralizzata BIC

In fine abbiamo valutato l'equivalenza markoviana dei due DAG:

```
> cat("I␣due␣DAG␣moralizzati␣sono␣equivalenti␣markovianamente?", equivalenza)

I due DAG moralizzati sono equivalenti markovianamente? FALSE
```

Per concludere questa sezione abbiamo effettuato un'analisi tramite **d-separation** per esplorare le relazioni condizionali tra le variabili. Questa tecnica consente di identificare indipendenze condizionate implicate dalla struttura del grafo. In particolare, la d-separation ci permette di affermare se due o più variabili risultano statisticamente indipendenti. Il tutto può essere eseguito in R grazie alla funzione *dsep()*, che serve a verificare se fue variabili sono indipendenti condizionalmente a un insieme di altre variabili in una rete Bayesiana.

```
> #Test di tutte le D-separation
> vars <- names(GSS_cleaned)
>
> for (i in 1:length(vars)) {
+   for (j in 1:length(vars)) {
+     if (i < j) {
+       cat(vars[i], "-", vars[j], ":␣", dsep(model_bnstd_bic, vars[i], vars[j])
+     }
+   }
+ }
> for (i in 1:length(vars)) {
+   for (j in 1:length(vars)) {
```

```
+     for (k in 1:length(vars)) {
+       if (i < j && k != i && k != j) {
+         result <- dsep(model_bnstd_bic, x = vars[i], y = vars[j],
                                                    z = vars[k])
+         cat(vars[i], "   ", vars[j], "|", vars[k], ":", result, "\n")
+       }
+     }
+   }
+ }
```

Di seguito riportiamo le relazioni più interessanti, risultato del codice sopra:

```
#Confronto tra due variabili
CAPPUN - GUNLAW :  FALSE
CAPPUN - SEX :  FALSE
CAPPUN - ABRAPE :  FALSE
CAPPUN - CONFINAN :  FALSE
CAPPUN - SATJOB :  FALSE
GUNLAW - SEX :  FALSE
GUNLAW - ABRAPE :  FALSE
GUNLAW - CONFINAN :  FALSE
GUNLAW - SATJOB :  FALSE
SEX - ABRAPE :  FALSE
SEX - CONFINAN :  FALSE
SEX - SATJOB :  FALSE
ABRAPE - CONFINAN :  FALSE
ABRAPE - SATJOB :  FALSE
CONFINAN - SATJOB :  FALSE
#Confronto tra tre variabili
CAPPUN  CONFINAN | SEX : TRUE
CAPPUN  SATJOB | SEX : TRUE
CAPPUN  SATJOB | CONFINAN : TRUE
GUNLAW  ABRAPE | CAPPUN : TRUE
GUNLAW  CONFINAN | SEX : TRUE
GUNLAW  SATJOB | SEX : TRUE
GUNLAW  SATJOB | CONFINAN : TRUE
SEX  ABRAPE | CAPPUN : TRUE
SEX  SATJOB | CONFINAN : TRUE
ABRAPE  CONFINAN | CAPPUN : TRUE
ABRAPE  CONFINAN | SEX : TRUE
ABRAPE  SATJOB | CAPPUN : TRUE
ABRAPE  SATJOB | SEX : TRUE
ABRAPE  SATJOB | CONFINAN : TRUE
```

I risultati mostrano le d-separation tra coppie di variabili principali risultano quasi tutte false, sia marginalmente sia condizionando altre variabili. Questo suggerisce che il modello apprende una forte dipendenza globale tra le opinioni espresse dagli individui su temi diversi. In particolare,

conoscere il sesso dell'individuo permette di spezzare molte dipendenza tra variabili come l'opinione sulla pena di morte, la soddisfazione lavorativa e la fiducia nelle imprese. Analogamente, la variabile *CAPPUN*, si comporta come una variabile informativa, capace di spiegare o interrompere diverse relazioni. Questo significa che alcune connessioni apparenti tra variabili possono essere attribuite esclusivamente all'effetto condiviso esercitato da una terza variabile. L'analisi delle d-separation in questo contesto si rivelo uno strumento potente in grado di comprendere la struttura causale appresa e identificare potenziali relazioni spurie.

5.1.2 Variabili di background

Analizzando bene la figura 13 ci rendiamo conto di come questo approccio standard presenti relazioni poco plausibili dal punto di vista causale. Per esempio la relazione tra *SEX* e *GUNLAW* sembra indicare che il sesso di un individuo sia dipendente in qualche modo dalla sua opinione rispetto al porto d'armi. Per risolvere questo problema è necessario specificare un ordine alle variabili suddividendole in:

- **Variabili di background**: *SEX*, *SATJOB* e *CONFINAN* sono le variabili che descrivono informazioni o caratteristiche riguardanti gli individui, che non sono però il diretto soggetto dell'analisi.
- **Variabili target**: *CAPPUN*, *GUNLAW* e *ABRAPE*, sono le variabili che racchiudono l'opinionde dell'individuo e di cui ne vogliamo studiare il valore rispetto alle altre. Nel nostro caso di studio stiamo valutando la variabile *CAPPUN*, per cui sarà questa la variabile analizzata.

```
>#Selezione delle variabili di background e target
> backgnd_vars <- c("SEX", "SATJOB", "CONFINAN")
> target_vars <- c("CAPPUN", "GUNLAW", "ABRAPE")
>#Creazione della blacklist
> blacklist <- expand.grid(from = target_vars, to = backgnd_vars)
>#Creazione del modello bayesiano con variabili target specificate
> target_model_bn <- hc(GSS_cleaned, blacklist = blacklist)
> dag_target_bn <- as.igraph(target_model_bn)
> graphviz.plot(target_model_bn)
```

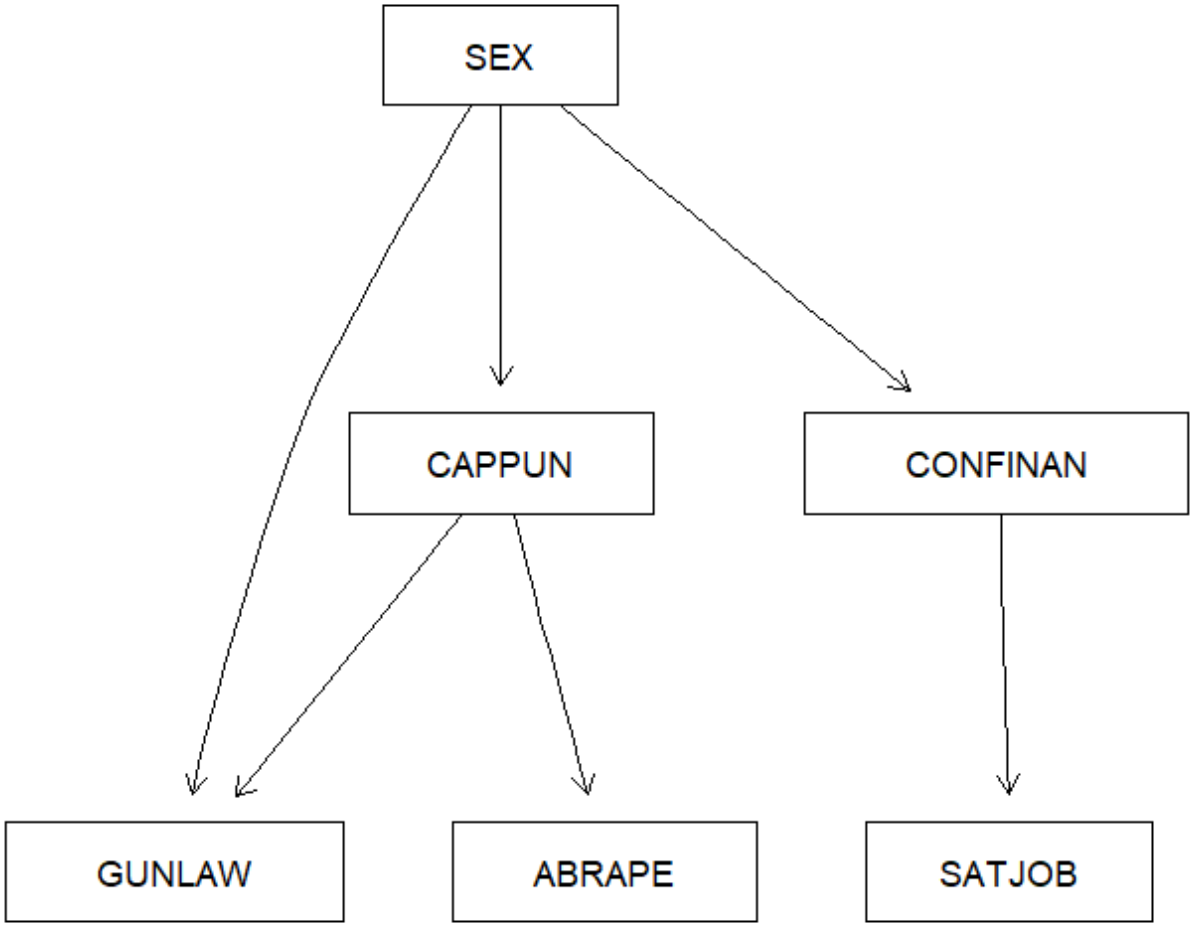

Figura 18: Rete Bayesiana con variabili target

Con questo procedimento siamo riusciti a raggiungere una rete che presenta degli archi più consoni rispetto al dominio di analisi. Oltre a invertire la relazione che volevamo correggere, notiamo come si sia invertita anche la relazione tra *CAPPUN* e *GUNLAW*. Tornando alla figura possiamo sviluppare un ordinamento topologico delle variabili come segue:

$$X_{SEX} \prec X_{CONFINAN} \prec X_{SATJOB} \prec X_{CAPPUN} \prec X_{GUNLAW} \prec X_{ABRAPE} \tag{4}$$

5.2 Stima dei parametri

Una volta che abbiamo costruito la rete Bayesiana possiamo stimare i parametri, ovvero calcolare per ogni variabile le sue relazioni di indipendenza rispetto ai genitori all'interno del DAG:

$$X_{SATJOB} \perp\!\!\!\perp X_{SEX}, X_{GUNLAW}, X_{CAPPUN}, X_{ABRAPE} | X_{CONFINAN} \tag{5}$$

$$X_{CONFINAN} \perp\!\!\!\perp X_{GUNLAW}, X_{CAPPUN}, X_{ABRAPE} | X_{SEX} \tag{6}$$

$$X_{GUNLAW} \perp\!\!\!\perp X_{CONFINAN}, X_{SATJOB} | X_{SEX} \tag{7}$$

$$X_{CAPPUN} \perp\!\!\!\perp X_{CONFINAN}, X_{SATJOB} | X_{SEX} \tag{8}$$

$$X_{ABRAPE} \perp\!\!\!\perp X_{SEX}, X_{GUNLAW}, X_{CONFINAN}, X_{SATJOB} | X_{CAPPUN} \tag{9}$$

Una volta costruite queste relazioni possiamo calcolare la probabilità congiunta totale.

5.3 Inferenza e interrogazione della rete

Una volta costruita la rete possiamo utilizzare la funzione *fit()* per trovare le probabilità marginali, congiunte e condizionali:

```
#Stima delle distribuzioni condizionate per ogni variabile
> bn_fit <- bn.fit(target_model_bn, data=GSS_cleaned)
> #Immissione in un oggetto grain
> bn_grain <- as.grain(bn_fit)
> #Interrogazione della rete
> #Marginali
> querygrain(bn_grain, nodes = "CAPPUN", type="marginal")

CAPPUN
Favorevole  Contrario
 0.7230428  0.2769572

> #Congiunta per CONFINAN e SATJOB
> querygrain(bn_grain, nodes = c("CONFINAN", "SATJOB"), type="joint")

           SATJOB
CONFINAN    Soddisfatto   Neutrale Insoddisfatto
  Fiducioso  0.15665417 0.09841586    0.03535624
  Neutrale   0.25881993 0.21795362    0.08196219
  Scettico   0.06397796 0.05632509    0.03053494

> #Condizionata di CAPPUN dato SEX
> querygrain(bn_grain, nodes = c("CAPPUN", "SEX"), type="conditional")

            SEX
CAPPUN        Maschio  Femmina
  Favorevole 0.773608 0.683565
  Contrario  0.226392 0.316435
```

Possiamo notare come il 22.6% dei maschi è contrario alla pena di morte, lo stesso valore che avevamo riscontrato nel calcolo della matrice di contingenza tra le due variabili. Il calcolo dei coefficienti basato sui log-odds, accennato in precedenza, viene calcolato come segue:

$$logit(P(CAPPUN|SEX)) = \beta_0 + \beta_1 \cdot X_{SEX} \tag{10}$$

Dalla tabella di contingenza:

- β_0 rappresenta i log-odds dei contrari alla pena di morte che sono maschi.
- β_1 rappresenta la differenza tra i log-odds dei maschi contrari e le femmine contrarie.

Per prima cosa si calcola la probabilità di essere favorevoli alla pena dato l'essere maschio:

$$P(X_{CAPPUN} = Contrario | X_{SEX} = Maschio) \tag{11}$$

Riprendendo i valori numerici calcolati durante le matrici di contingenza ricaviamo

$$P(X_{CAPPUN} = Contrario | X_{SEX} = Maschio) = \frac{1297}{4432+1296} = 0.2264 \tag{12}$$

Per calcolare β_0 è sufficiente calcolare la probabilità di essere contrario dato l'essere maschio, diviso l'evento complementare tutto dentro ad un logaritmo per il calcolo del logit:

$$\beta_0 = log\left(\frac{P(Contrario|Maschio)}{1-P(Contrario|Maschio)}\right) = -1.229 \tag{13}$$

Per β_1 invece calcoliamo:

$$log\left(\frac{P(Contrario|Femmina)}{1-P(Contrario|Femmina)}\right) = -0.770 \tag{14}$$

e calcolando la differenza tra questo valore e β_0 si può ricavare:

$$\beta_1 = -0.770 + 1.229 = 0.459 \tag{15}$$

5.4 Analisi con evidenze osservate

In una rete Bayesiana, un'evidenza è un informazione osservata su una o più variabili del modello. Nella pratica è come dare per certo che una variabile ha un determinato valore scelto. Abbiamo visto come le reti Bayesiane sono modelli probabilistici che rappresentano dipendenze condizionali tra le variabili analizzate, quando viene data un'evidenza la rete aggiorna le probabilità posteriori delle altre variabili in base a questa informazione.
Grazie al pacchetto *Rgraphviz*, possiamo visualizzare la rete bayesiana con al suo interno le distribuzioni di probabilità marginali per ogni variabile:

```
>#Visualizzazione della BN con distribuzione
> graphviz.chart(bn_fit,
+                type = "barprob",
+                bar.col = "brown4",
+                )
>
```

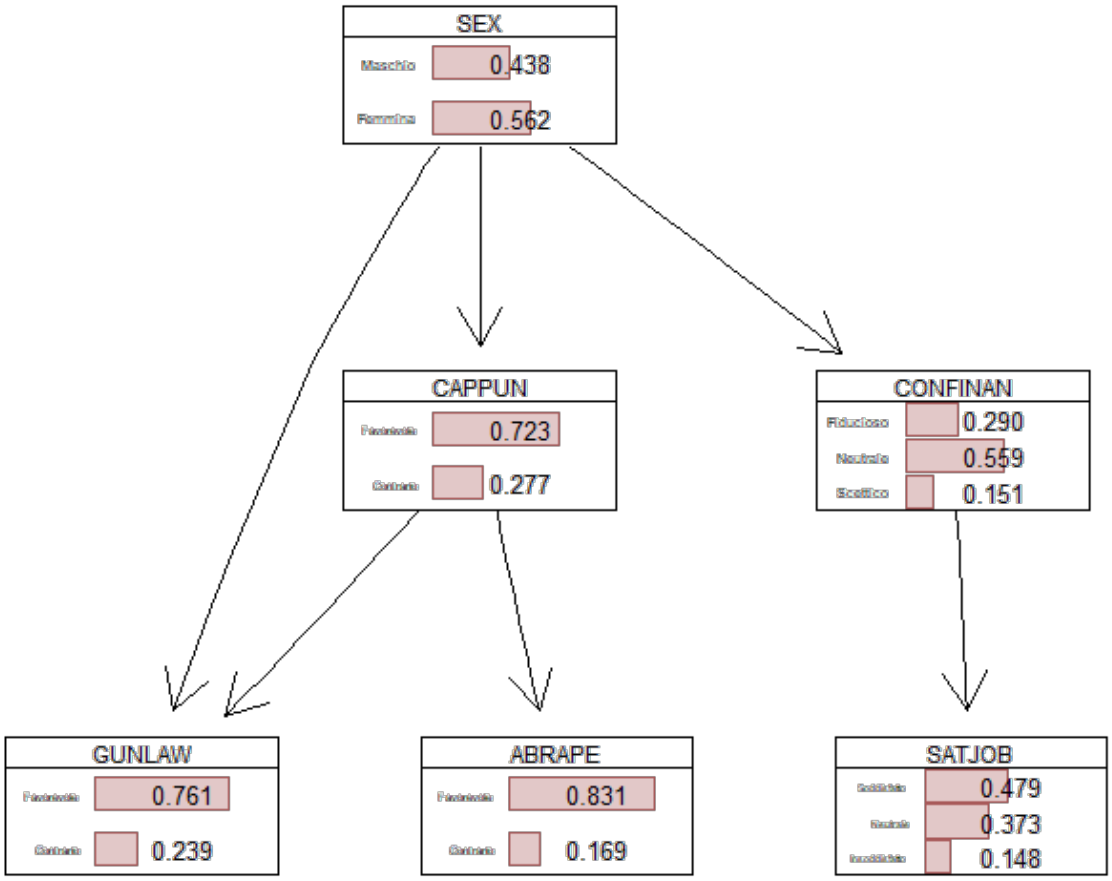

Figura 19: Rete Bayesiana con probabilità marginali

La variabile *SEX* sembra essere quella che influenza più variabili, per cui quello che andremo a fare è visualizzare la rete Bayesiana quando SEX = Maschio e quando è uguale a Femmina, di modo da poter capire con quale probabilità si ha un'opinione o meno.

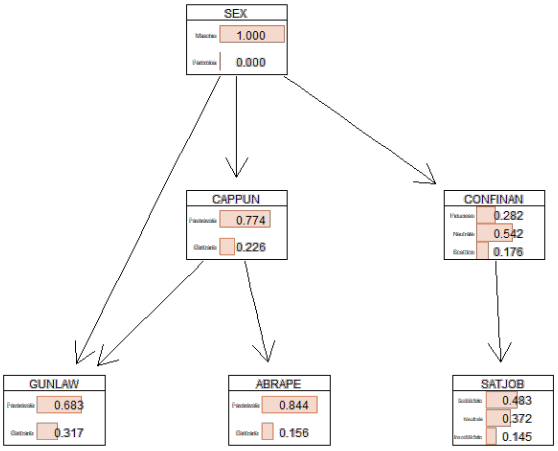

Figura 20: Rete Bayesiana con evidenza Maschio

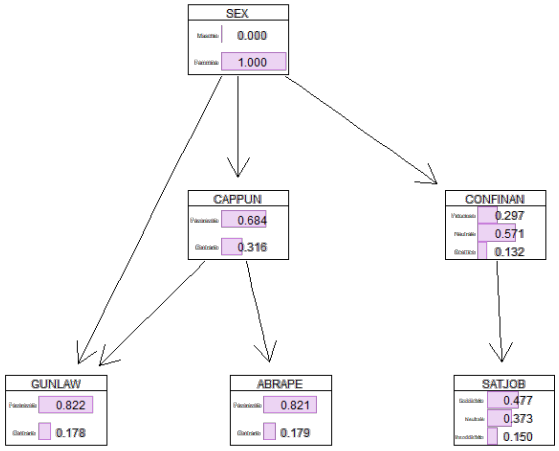

Figura 21: Rete Bayesiana con evidenza Maschio

Dall'analisi condotta tramite evidenza, emergono alcune differenze significative nei profili probabilistici tra individui si sesso maschile e femminile:

- **Sesso maschile**: L'appartenenza al sesso maschile è associata ad un aumento della probabilità di essere favorevoli alla pena di morte mentre diminuisce la probabilità di essere favorevoli ad una legge sul porto d'armi. Le altre variabili considerate, non mostrano varianze significative.
- **Sesso femminile**: Per le donne invece, si osserva una minore probabilità di essere favorevoli alla pena di morte, ma una maggiore propensione ad approvare una legge sul porto d'armi, superando quella osservata per gli uomini. È inoltre presente una lieve diminuzione nella probabilità di essere favorevoli alla legge sull'aborto.

Possiamo dire che il genere sembra influenzare in modo selettivo alcune opinioni sociali e politiche. I maschi tendono ad avere un approccio più punitivo e repressivo, mentre le donne si mostrano generalmente più contrarie a tale misura. Tuttavia, le donne suggeriscono un atteggiamento più orientato all'autodifesa.

Nella seguente analisi dell'evidenza abbiamo suddiviso la rete Bayesiana tra favorevoli e contrari alla pena di morte:

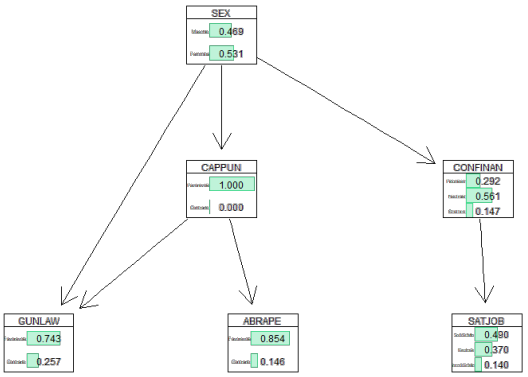

Figura 22: Rete Bayesiana con evidenza Favorevole alla pena di morte

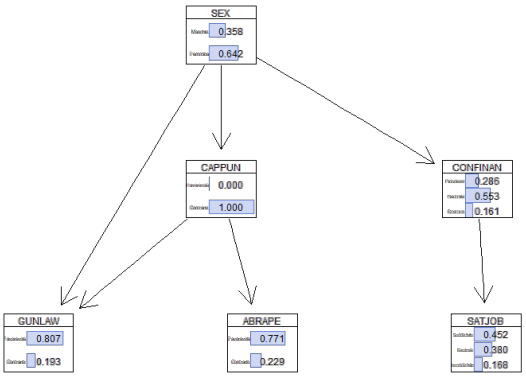

Figura 23: Rete Bayesiana con evidenza Contraria alla pena di morte

Dall'analisi condotta con questa evidenza emerge che:

- **Favorevoli alla pena di morte**: Gli individui favorevoli alla pena di morte risultano avere una probabilità più alta di essere favorevoli alla legge sull'aborto ed una più bassa nell'essere soddisfatti del luogo di lavoro.
- **Contrari alla pena di morte**: Notiamo come chi è contro la pena di morte ha una probabilità più alta di essere favorevole alla legge sul porto d'armi, con una grande diminuzione dell'essere favorevole alla legge sull'aborto. Inoltre si nota un lieve peggioramento nella probabilità di avere fiducia nelle imprese e dell'essere soddisfatti sul posto di lavoro.

Possiamo dire che gli individui favorevoli alla pena di morte sembrano presentare un profilo più coerente con una visione più autoritaria e normativa. Chi invece è contro la pena di morte, mostra un profilo meno omogeneo, dove pur essendo più favorevole al porto d'armi, manifesta una ridotta accettazione dell'aborto e una diminuzione della fiducia economica.

Come ultimo test di evidenza singola andiamo a valutare come varia la probabilità per chi è favorevole o contrario alla legge sull'aborto:

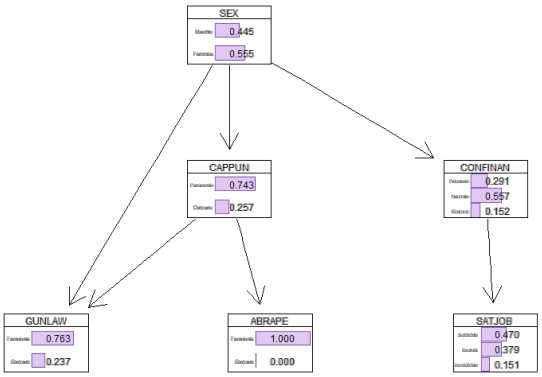

Figura 24: Rete Bayesiana con evidenza Favorevole alla legge sull'aborto

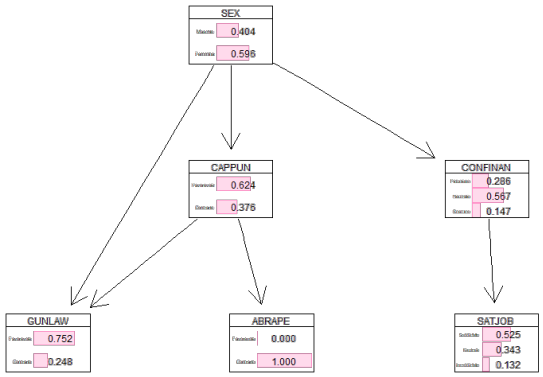

Figura 25: Rete Bayesiana con evidenza Contraria alla legge sull'aborto

Con questa evidenza possiamo dire che:

- **Favorevoli alla legge sull'aborto**: Per chi è favorevole a questa legge notiamo come sia più probabile di essere insoddisfatti della propria situazione sul lavoro e di come aumenti leggermente la probabilità di essere favorevoli alla pena di morte.
- **Contrari alla legge sull'aborto**: Per quanto riguarda gli individui contrari alla legge sull'aborto possiamo notare come la probabilità che siano favorevoli alla pena di morte cali di molto, insieme ad un leggero calo della probabilità di essere favorevoli alla legge sul proto d'armi. Notiamo inoltre una maggiore soddisfazione sul lavoro ed una leggera diminuzione nella fiducia delle imprese.

Questa evidenza suggerisce come l'orientamento verso la legge sull'aborto possa riflettere un insieme più ampio di atteggiamenti sociali e morali. In particolare, chi è contrario sembra esprimere una visione più conservatrice ma più soddisfatta a livello personale, mentre i favorevoli sembrano più progressisti su temi etici, ma più critici rispetto alla propria condizione lavorativa.

Passiamo adesso all'analisi dell'evidenza su più variabili, di seguito troviamo le configurazione per le variabili *CAPPUN* e *SEX*:

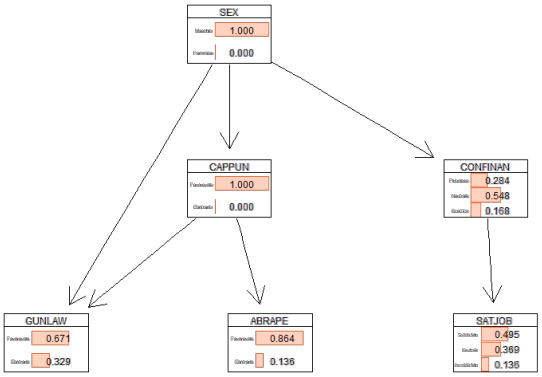

Figura 26: Rete Bayesiana con evidenza Maschio e Favorevole alla pena

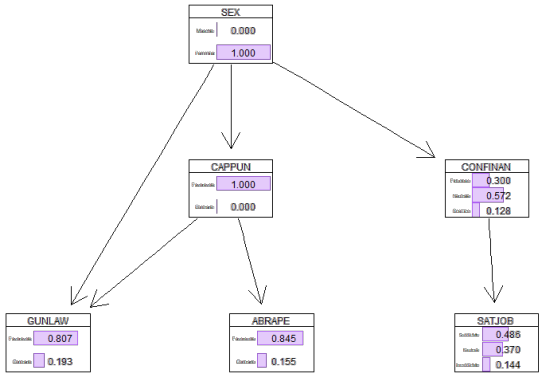

Figura 27: Rete Bayesiana con evidenza Femmina e Favorevole alla pena

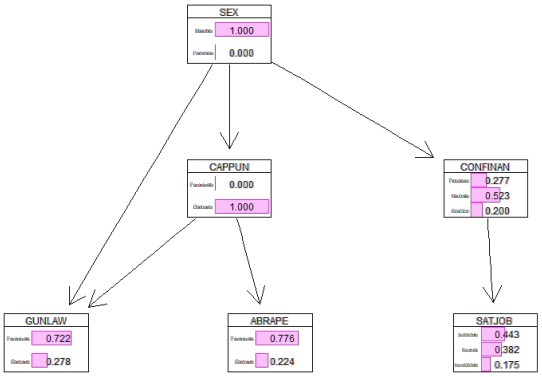

Figura 28: Rete Bayesiana con evidenza Maschio e Contrario alla pena

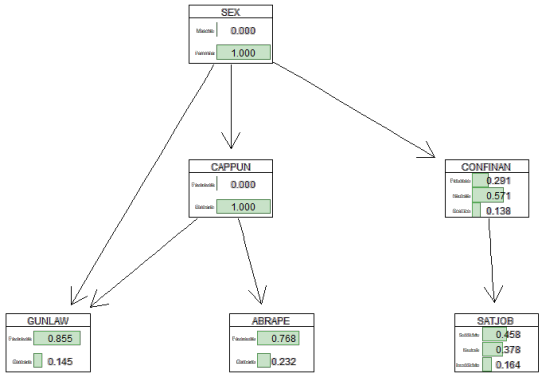

Figura 29: Rete Bayesiana con evidenza Femmina e Contrario alla pena

Da queste evidenze condizionate possiamo capire che i maschi favorevoli alla pena di morte hanno una probabilità più bassa di essere favorevoli alla legge sul porto d'armi ma ne hanno una più alta nell'essere favorevoli alla legge sull'aborto, possiamo inoltre notare un leggero scetticismo verso l'industria. Per quanto riguarda le femmine che sono favorevoli, possiamo notare come ci sia una probabilità più altra nell'essere favorevole ad una legge sul porto d'armi ed anche nell'essere favorevole alla legge sull'aborto, anche le variabili economiche risultano avere probabilità più alte. Passando agli individui che sono contrari possiamo dire che i maschi tendono ad essere leggermente meno favorevoli alla legge sul porto d'armi e lo stesso vale per la probabilità di essere favorevoli alla legge sull'aborto che scende, questi individui tendono anche ad avere meno fiducia nell'industria e ad essere più insoddisfatti del posto di lavoro. Per quanto riguarda le femmine che sono contrarie, possiamo dire che aumenta la probabilità di essere favorevoli alla legge sul porto d'armi a discapito

dell'essere favorevoli alla legge sull'aborto, notiamo inoltre una leggere insoddisfazione del posto di lavoro.

5.5 Selezione di un modello predittivo

In questa sezione andremo a stimare un modello di regressione logistica, abbandonando la logica puramente grafica vista fin'ora, con lo scopo di analizzare i fattori che influenzano la probabilità che un individuo sia favorevole alla nostra variabile target *GUNLAW*. La regressione logistica è una tecnica appropriata per modellare una variabile dipendente binaria, come nel nostro caso, in funzione di un insieme di variabile covariate esplicative.
Per la stima del modello utilizzeremo la funzione *glm()* (generalized linear model) specificando la famiglia binomiale, in quanto la variabile risposta può assumere solo due valori.
Successivamente applicheremo la funzione *step()* per effettuare una selezione automatica delle variabili da includere nel modello, utilizzando il criterio AIC o BIC come metrica di decisione. Come abbiamo già visto il procedimento di selezione può essere:

- **Forward**: si parte da un modello nullo e si aggiungono progressivamente le variabile più significative.
- **Backward**: si parte da un modello completo e si rimuovono le variabili meno significative.
- **Both**: combinazione dei due approcci.

La funzione *step()* ci consente di individuare un modello parsimonioso, questo è fondamentale perché la scelta di un modello con troppe poche variabili comporterebbe un'alta distorsione e quindi il rischio di underfitting, con una scarsa spiegabilità dei dati. Scegliere invece un modello troppo complesso e quindi con troppe variabili potrebbe portare ad un alta varianza e quindi al rischio di overfitting, ovvero un elevato adattamento ai dati e bassa generalizzazione.
Definiamo il modello nullo e quello saturo per poter eseguire i due approcci:

```
>#Definizione del modello nullo e saturo
> null_model <- glm(GUNLAW ~ 1, data = GSS_cleaned, family = binomial)
> full_model <- glm(GUNLAW ~.^2, data = GSS_cleaned, family = binomial)
>#Definizioni dello scope per definire gli intervalli di lavoro
> scope <- list(lower=formula(null_model), upper = formula(full_model))
>
> back_model_AIC <- step(full_model, scope = scope, direction = "backward", k =
>
> summary(back_model_AIC)


Call:
glm(formula = GUNLAW ~ CAPPUN + SEX + ABRAPE + SATJOB + SEX:SATJOB,
    family = binomial, data = GSS_cleaned)

Coefficients:
                            Estimate Std. Error z value Pr(>|z|)
(Intercept)                 -0.79284    0.04336 -18.286  < 2e-16 ***
```

```
CAPPUNContrario                -0.30715    0.04932  -6.227 4.74e-10 ***
SEXFemmina                     -0.64376    0.06041 -10.657  < 2e-16 ***
ABRAPEContrario                 0.13562    0.05544   2.446  0.01443 *
SATJOBNeutrale                  0.16141    0.06192   2.607  0.00914 **
SATJOBInsoddisfatto             0.07595    0.08558   0.887  0.37483
SEXFemmina:SATJOBNeutrale      -0.23576    0.09127  -2.583  0.00979 **
SEXFemmina:SATJOBInsoddisfatto -0.09027    0.12435  -0.726  0.46790
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14369  on 13066  degrees of freedom
Residual deviance: 13976  on 13059  degrees of freedom
AIC: 13992

Number of Fisher Scoring iterations: 4
```

Possiamo utilizzare il metodo di penalizzazione BIC impostando $k = log(nrow)$:

```
> back_model_BIC <- step(full_model, scope = scope, direction = "backward", k =
> summary(back_model_BIC)

Call:
glm(formula = GUNLAW ~ CAPPUN + SEX, family = binomial, data = GSS_cleaned)

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)    -0.70249    0.03018 -23.280  < 2e-16 ***
CAPPUNContrario -0.29344    0.04900  -5.988 2.12e-09 ***
SEXFemmina     -0.74258    0.04189 -17.727  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14369  on 13066  degrees of freedom
Residual deviance: 13989  on 13064  degrees of freedom
AIC: 13995

Number of Fisher Scoring iterations: 4
```

Passiamo ora alla costruzione del modello in direzione *forward* specificandolo nella funzione *step()*:

```
 > forward_model_AIC <- step(null_model, scope = scope, direction = "forward",
k = 2)
 > summary(forward_model_AIC)
```

```
Call:
glm(formula = GUNLAW ~ SEX + CAPPUN + ABRAPE, family = binomial,
    data = GSS_cleaned)

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)     -0.72072    0.03119 -23.111  < 2e-16 ***
SEXFemmina      -0.74515    0.04192 -17.777  < 2e-16 ***
CAPPUNContrario -0.30460    0.04926  -6.184 6.25e-10 ***
ABRAPEContrario  0.13090    0.05535   2.365    0.018 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14369  on 13066  degrees of freedom
Residual deviance: 13984  on 13063  degrees of freedom
AIC: 13992

Number of Fisher Scoring iterations: 4
```

Stessa cosa per il modello BIC:

```
> forward_model_BIC <- step(null_model, scope = scope, direction = "forward", k
> summary(forward_model_BIC)


Call:
glm(formula = GUNLAW ~ SEX + CAPPUN, family = binomial, data = GSS_cleaned)

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)     -0.70249    0.03018 -23.280  < 2e-16 ***
SEXFemmina      -0.74258    0.04189 -17.727  < 2e-16 ***
CAPPUNContrario -0.29344    0.04900  -5.988 2.12e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14369  on 13066  degrees of freedom
Residual deviance: 13989  on 13064  degrees of freedom
AIC: 13995

Number of Fisher Scoring iterations: 4
```

Aggiungiamo per completezza anche la costruzione con il metodo *both*:

```
>both_model_AIC <- step(null_model, scope = scope, direction = "both", k = 2)
>summary(both_model_AIC)

Call:
glm(formula = GUNLAW ~ SEX + CAPPUN + ABRAPE, family = binomial,
    data = GSS_cleaned)

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)     -0.72072    0.03119 -23.111  < 2e-16 ***
SEXFemmina      -0.74515    0.04192 -17.777  < 2e-16 ***
CAPPUNContrario -0.30460    0.04926  -6.184 6.25e-10 ***
ABRAPEContrario  0.13090    0.05535   2.365    0.018 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14369  on 13066  degrees of freedom
Residual deviance: 13984  on 13063  degrees of freedom
AIC: 13992

Number of Fisher Scoring iterations: 4

>both_model_BIC <- step(null_model, scope = scope, direction = "both", k = log(n
>summary(both_model_BIC)

Call:
glm(formula = GUNLAW ~ SEX + CAPPUN, family = binomial, data = GSS_cleaned)

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)     -0.70249    0.03018 -23.280  < 2e-16 ***
SEXFemmina      -0.74258    0.04189 -17.727  < 2e-16 ***
CAPPUNContrario -0.29344    0.04900  -5.988 2.12e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14369  on 13066  degrees of freedom
Residual deviance: 13989  on 13064  degrees of freedom
AIC: 13995

Number of Fisher Scoring iterations: 4
```

Dopo aver stimato diversi modelli di regressione logistica, con variabile target *GUNLAW*, possiamo valutare i valori del p-value per ogni variabile presente nel modello risultante.
Notiamo come la selezione secondo BIC, più conservativa, ha portato ad un modello parsimonioso in cui le uniche variabili rilevanti risultano essere il sesso e l'opinione sulla pena di morte. In particolare:

- Le donne mostrano una minore propensione a supportare una legge sul porto d'armi rispetto agli uomini.
- Gli individuo contrari alla pena di morte hanno anch'essi una probabilità inferiore di essere favorevoli al porto d'armi.

I modelli più complessi, secondo AIC, includono anche l'opinione sulla legge dell'aborto, che risulta avere un effetto più debole ma comunque significativo.
Complessivamente, il modello selezionato tramite BIC appare il più parsimonioso ed efficace, indicando che le posizioni sui temi etici e di genere siano i principali predittori dell'opinione sulla legge sul porto d'armi

6 Conclusioni

In questo lavoro abbiamo analizzato le relazioni tra opinioni su temi etici e caratteristiche individuali a partire dal dataset GSS, focalizzandoci sulla variabile *GUNLAW*.
In una prima fase, abbiamo costruito dei grafi indiretti, che si basano sull'associazione tra le variabili, che ci hanno permesso di identificare in modo visivo e rapido le relazioni più forti. Questi grafi, seppur non orientati né causali, sono utili per una prima esplorazione della struttura dati analizzata, fornendo uno spunto per le successive analisi.
Come secondo passo, abbiamo stimato una rete Bayesiana, ottenendo una rappresentazione grafica direzionata delle dipendenza probabilistiche tra le variabili. Attraverso l'evidenza, sia su singole variabili che su variabili multiple, abbiamo osservato come gli individui di sesso maschile tendano ad avere un approccio più punitivo, mostrando una maggiore probabilità di supportare la pena di morte, mentre le donne sembrano generalmente più contrarie a questa misura. Inoltre, gli uomini risultano più favorevoli alla legge sul porto d'armi, anche se con alcune sfumature rilevate nelle evidenze condizionate. L'analisi condizionata su altre variabili ha mostrato interessanti relazioni come:

- Chi è favorevole alla pena di morte tende anche a essere più favorevole alla legge sull'aborto, ma meno soddisfatto del proprio lavoro.

- Chi è contrario alla legge sull'aborto mostra una minore propensione alla pena di morte e una maggiore soddisfazione lavorativa.

Successivamente, è stata effettuata un'analisi di regressione logistica con *GUNLAW* come variabile risposta, per individuare i principali predittori dell'opinione sulla legge sul porto d'armi. Abbiamo applicato la selezione automatica delle variabili tramite i criteri AIC e BIC, con approcci forward e backward. Dai risultati possiamo trarre che:

- Il modello più parsimonioso secondo il BIC include solo le variabili SEX e CAPPUN, indicando che: Le donne hanno una significativa probabilità più bassa di supportare una legge sul porto d'armi.

- AIC ha suggerito modelli leggermente più complessi, includendo ad esempio anche l'opinione sulla legge sull'aborto, ma senza sostanziale miglioramento in termini di devianza residua.

In sintesi, il genere e l'orientamento etico verso la pena di morte emergono come i principali determinanti dell'opinione su una possibile legge sul porto d'armi, confermando come i temi morali e identitari siano fortemente intrecciati nelle scelte e opinioni politiche e sociali degli individui studiati.

7 Appendice

Di seguito viene riportato il codice completo in R, utilizzato per l'analisi in questione. Il tutto può essere anche visualizzato all'interno della repository su Github.

```
library(gRim)
library(gRain)
library(gRbase)
library(ggplot2)
library(bnlearn)
library(igraph)
library(Rgraphviz)

# Caricamento del dataset
load("GSS.RData")
# Visualizzazione delle variabili contenute nel DS
# insieme al sommario statistico
summary(GSS)
#Visualizzazione dela struttura del DS
str(GSS)


#Eliminazione dei valori mancanti/non validi
GSS_cleaned <- na.omit(GSS)
#Visualizzazione del DS pulito
str(GSS_cleaned)

#Trasformazione delle variabili in fattori
#con etichette per migliorare la leggibilit

#CAPPUN
GSS_cleaned$CAPPUN <- factor(GSS_cleaned$CAPPUN,
                             levels = c(1, 2),
                             labels = c("Favorevole", "Contrario"))

# GUNLAW
GSS_cleaned$GUNLAW <- factor(GSS_cleaned$GUNLAW,
                             levels = c(1, 2),
                             labels = c("Favorevole", "Contrario"))

# SEX
GSS_cleaned$SEX <- factor(GSS_cleaned$SEX,
                          levels = c(1, 2),
                          labels = c("Maschio", "Femmina"))

# ABRAPE
GSS_cleaned$ABRAPE <- factor(GSS_cleaned$ABRAPE,
                             levels = c(1, 2),
```

```
                              labels = c("Favorevole", "Contrario"))

# CONFINAN
GSS_cleaned$CONFINAN <- factor(GSS_cleaned$CONFINAN,
                               levels = c(1, 2, 3),
                               labels = c("Fiducioso",
                                          "Neutrale",
                                           "Scettico"))

# SATJOB
GSS_cleaned$SATJOB <- factor(GSS_cleaned$SATJOB,
                             levels = c(1, 2, 3),
                             labels = c("Soddisfatto",
                                        "Neutrale",
                                        "Insoddisfatto"))
#Visualizzazione del DS dopo la fattorizzazione
str(GSS_cleaned)



#Visualizzaione grafico a barre

ggplot(GSS_cleaned, aes(x = GUNLAW)) +
  geom_bar(fill = "purple3") +
  geom_text(stat = "count", aes(label = ..count..), vjust = -0.5) +
  labs(title = "Distribuzione di GUNLAW",
       x = "Opinione", y = "Frequenza") +
  theme_minimal()



#Selezione delle variabili di confronto
variabili_confronto <- c("SEX", "SATJOB", "ABRAPE", "GUNLAW", "CONFINAN")
#Ciclo per visualizzazione della matrice di contingenza per ogni variabile
for (var in variabili_confronto) {

  tab <- table(GSS_cleaned$CAPPUN, GSS_cleaned[[var]])
  dimnames(tab)[[1]] <- levels(GSS_cleaned$CAPPUN)
  dimnames(tab)[[2]] <- levels(GSS_cleaned[[var]])
  names(dimnames(tab)) <- c("CAPPUN", var)

  #Matrice totale
  cat("\n=======================================\n")
  cat("Matrice di contingenza assoluta: CAPPUN vs", var, "\n")
  cat("=======================================\n")
  print(tab)

  #Matrice percentuale
```

```
  prop_tab <- round(prop.table(tab, margin = 2) * 100, 1)

  cat("\n----------------------------------------\n")
  cat("Distribuzione % per colonna: CAPPUN vs", var, "\n")
  cat("----------------------------------------\n")
  print(prop_tab)
}

#Visualizzazione heatmap
for (var in variabili_confronto) {

  # Tabella di contingenza
  tab <- table(GSS_cleaned$CAPPUN, GSS_cleaned[[var]])

  # Calcolo percentuali colonna per colonna
  tab_perc <- round(prop.table(tab, margin = 2) * 100, 1)

  # Conversione
  df_heat <- as.data.frame(tab_perc)
  names(df_heat) <- c("CAPPUN", "Variabile", "Percentuale")
  df_heat$CAPPUN <- factor(df_heat$CAPPUN, levels = c("Contrario",
                                                      "Favorevole"))

  # Heatmap con percentuali
  print(
  ggplot(df_heat, aes(x = Variabile, y = CAPPUN, fill = Percentuale)) +
    geom_tile(color = "white") +
    geom_text(aes(label = paste0(Percentuale, "%")), color = "black", size = 4) +
    scale_fill_gradient(low = "white", high = "firebrick") +
    labs(title = paste("Heatmap percentuale: CAPPUN vs", var),
         x = var, y = "CAPPUN") +
    theme_minimal()
  )
}



#Creazione del modello saturo
model_sat <- dmod(~.^., GSS_cleaned)

#Creazione del modello di indipendenza
model_ind <- dmod(~.^1, GSS_cleaned)

#Grafo backward con penalizzazione AIC (Default)
backAIC_model <- stepwise(model_sat)
plot(backAIC_model)
title(main="UG backward AIC")
```

```
#Grafo backward con penalizzazione BIC
backBIC_model <- stepwise(model_sat, k = log(nrow(GSS_cleaned)))
plot(backBIC_model)
title(main="UG backward BIC")

#Grafo forward con penalizzazione AIC
forwardAIC_model <- stepwise(model_ind, direction="forward")
plot(forwardAIC_model)
title(main="UG forward AIC")

#Grafo forward con penalizzazione BIC
forwardBIC_model <- stepwise(model_ind, k = log(nrow(GSS_cleaned)),
                                                direction="forward")
plot(forwardBIC_model)
title(main="UG forward BIC")


#Grafo both con penalizzazione AIC
bothAIC_model <- stepwise(model_ind, direction="both")
plot(bothAIC_model)
title(main = "UG both AIC")

#Grafo both con penalizzazione BIC
bothBIC_model <- stepwise(model_ind, k = log(nrow(GSS_cleaned)),
                                                direction = "both" )
plot(bothBIC_model)
title(main = "UG both BIC")


#Grafi diretti + Reti bayesiane

model_bnstd <- hc(GSS_cleaned)
dag_bnstd <-as.igraph(model_bnstd)
graphviz.plot(model_bnstd)

#Modelli con penalizzazione esplicita
model_bnstd_aic <- hc(GSS_cleaned, score="aic")
model_bnstd_bic <- hc(GSS_cleaned, score="bic")
graphviz.plot(model_bnstd_aic)
graphviz.plot(model_bnstd_bic)
graphviz.plot(moral(model_bnstd_aic))
graphviz.plot(moral(model_bnstd_bic))

equivalenza <- isTRUE(all.equal(moral(model_bnstd_aic),
                                        moral(model_bnstd_bic)))
cat("I due DAG moralizzati sono equivalenti markovianamente?", equivalenza)
```

```
#Test di tutte le D-separation sia per le coppie che per le condizionate
vars <- names(GSS_cleaned)

for (i in 1:length(vars)) {
  for (j in 1:length(vars)) {
    if (i < j) {
      cat(vars[i], "-", vars[j], ":␣", dsep(model_bnstd_bic, vars[i], vars[j]),
                                                                    "\n")
    }
  }
}

for (i in 1:length(vars)) {
  for (j in 1:length(vars)) {
    for (k in 1:length(vars)) {
      if (i < j && k != i && k != j) {
        result <- dsep(model_bnstd_bic, x = vars[i], y = vars[j], z = vars[k])
        cat(vars[i], "   ", vars[j], "|", vars[k], ":", result, "\n")
      }
    }
  }
}

#Rete Bayesiana con variabili target esplicite
backgnd_vars <- c("SEX", "SATJOB", "CONFINAN")
target_vars <- c("CAPPUN", "GUNLAW", "ABRAPE")
blacklist <- expand.grid(from = target_vars, to = backgnd_vars)
target_model_bn <- hc(GSS_cleaned, score="bic" ,blacklist = blacklist)
graphviz.plot(target_model_bn)
```

```
#Stima delle distribuzioni condizionate per ogni variabile
bn_fit <- bn.fit(target_model_bn, data=GSS_cleaned)
#Immissione in un oggetto grain
bn_grain <- as.grain(bn_fit)
#Interrogazione della rete
#Marginali
querygrain(bn_grain, nodes = "CAPPUN", type="marginal")
#Condizionate
querygrain(bn_grain, nodes = c("CAPPUN", "SEX"), type="conditional")
#Congiunta
querygrain(bn_grain, nodes = c("CONFINAN", "SATJOB"), type="joint")
```

```
graphviz.chart(bn_fit,
               type = "barprob",
               bar.col = "brown4",
               )

#Impostazione delle evidenze - SEX
GSS_male <- subset(GSS_cleaned, SEX == "Maschio")
bn_fit_male <- bn.fit(target_model_bn, data = GSS_male)
GSS_female <- subset(GSS_cleaned, SEX == "Femmina")
bn_fit_female <- bn.fit(target_model_bn, data = GSS_female)
graphviz.chart(bn_fit_male,
               type = "barprob",
               bar.col = "sienna3"
)
graphviz.chart(bn_fit_female,
               type = "barprob",
               bar.col = "mediumorchid3"
)

#Impostazione delle evidenze - CAPPUN
GSS_cappun_F <- subset(GSS_cleaned, CAPPUN == "Favorevole")
bn_fit_cappun_F <- bn.fit(target_model_bn, data = GSS_cappun_F)
GSS_cappun_C <- subset(GSS_cleaned, CAPPUN == "Contrario")
bn_fit_cappun_C <- bn.fit(target_model_bn, data = GSS_cappun_C)
graphviz.chart(bn_fit_cappun_F,
               type = "barprob",
               bar.col = "springgreen3"
)
graphviz.chart(bn_fit_cappun_C,
               type = "barprob",
               bar.col = "royalblue3"
)

#Impostazione delle evidenze - ABRAPE
GSS_abrape_F <- subset(GSS_cleaned, ABRAPE == "Favorevole")
bn_fit_abrape_F <- bn.fit(target_model_bn, data = GSS_abrape_F)
GSS_abrape_C <- subset(GSS_cleaned, ABRAPE == "Contrario")
bn_fit_abrape_C <- bn.fit(target_model_bn, data = GSS_abrape_C)
graphviz.chart(bn_fit_abrape_F,
               type = "barprob",
               bar.col = "purple3"
)
graphviz.chart(bn_fit_abrape_C,
               type = "barprob",
               bar.col = "hotpink"
)
```

```
#Impostazione delle evidenze - CAPPUN + SEX
GSS_cappun_F_SEX_M <- subset(GSS_cleaned, CAPPUN == "Favorevole" &
                                                    SEX == "Maschio")
bn_fit_cappun_F_SEX_M <- bn.fit(target_model_bn, data = GSS_cappun_F_SEX_M)
graphviz.chart(bn_fit_cappun_F_SEX_M,
               type = "barprob",
               bar.col = "orangered1"
)
GSS_cappun_F_SEX_F <- subset(GSS_cleaned, CAPPUN == "Favorevole" &
                                                    SEX == "Femmina")
bn_fit_cappun_F_SEX_F <- bn.fit(target_model_bn, data = GSS_cappun_F_SEX_F)
graphviz.chart(bn_fit_cappun_F_SEX_F,
               type = "barprob",
               bar.col = "purple2"
)
GSS_cappun_C_SEX_M <- subset(GSS_cleaned, CAPPUN == "Contrario" &
                                                    SEX == "Maschio")
bn_fit_cappun_C_SEX_M <- bn.fit(target_model_bn, data = GSS_cappun_C_SEX_M)
graphviz.chart(bn_fit_cappun_C_SEX_M,
               type = "barprob",
               bar.col = "magenta2"
)
GSS_cappun_C_SEX_F <- subset(GSS_cleaned, CAPPUN == "Contrario" &
                                                    SEX == "Femmina")
bn_fit_cappun_C_SEX_F <- bn.fit(target_model_bn, data = GSS_cappun_C_SEX_F)
graphviz.chart(bn_fit_cappun_C_SEX_F,
               type = "barprob",
               bar.col = "forestgreen"
)


#Regresione logistica
null_model <- glm(GUNLAW ~ 1, data = GSS_cleaned, family = binomial)
full_model <- glm(GUNLAW ~.^2, data = GSS_cleaned, family = binomial)
scope <- list(lower=formula(null_model), upper = formula(full_model))

back_model_AIC <- step(full_model, scope = scope, direction = "backward",
                                                      k = 2, trace = TRUE)
summary(back_model_AIC)

#Per il metodo BIC utilizziamo k = log(n)
back_model_BIC <- step(full_model, scope = scope, direction = "backward",
                                                 k = log(nrow(GSS_cleaned)))
summary(back_model_BIC)

forward_model_AIC <- step(null_model, scope = scope, direction = "forward",
                                                                  k = 2)
summary(forward_model_AIC)
```

```
forward_model_BIC <- step(null_model, scope = scope, direction = "forward",
                                                  k = log(nrow(GSS_cleaned)))
summary(forward_model_BIC)


both_model_AIC <- step(null_model, scope = scope, direction = "both", k = 2)
summary(both_model_AIC)

both_model_BIC <- step(null_model, scope = scope, direction = "both",
                                           k = log(nrow(GSS_cleaned)))
summary(both_model_BIC)
```