

L'ambiente *R* Applicazioni in Acustica Ambientale

P. Scordino, S. Bande, C. Ronchi

ARPA Piemonte

29 novembre 2018



Generalità sull'analisi dei dati

Grazie all'evoluzione tecnologica, che ha reso fruibile a basso costo l'hardware per la **memorizzazione** dei **dati** e l'hardware per l'**elaborazione** sempre più **veloce** degli stessi, oggi possiamo contare su una moltitudine di dati disponibili che ha reso necessario l'utilizzo di strumenti software potenti e flessibili per poter **estrarre informazioni utili in tempi ragionevoli**.

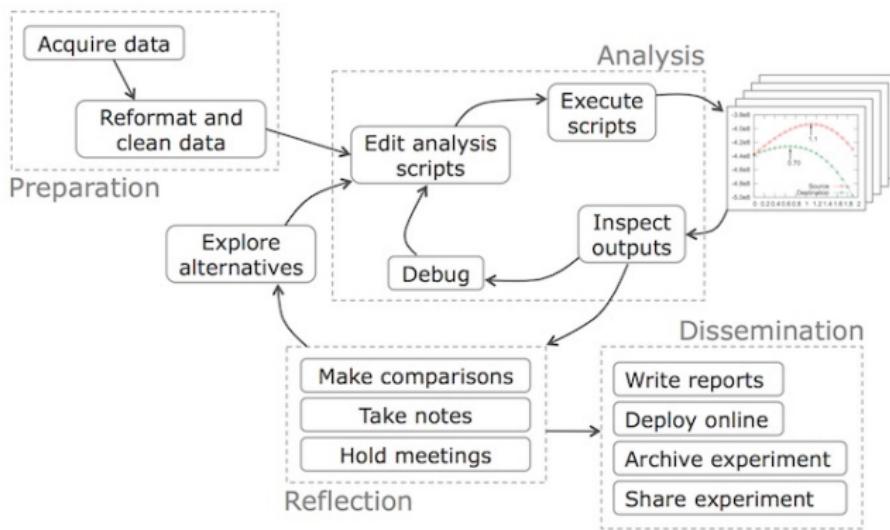
Generalità sull'analisi dei dati

Definizione di dati e informazione

- ▶ **Dato** dal latino *datum*, dono. I dati sono rappresentazioni di eventi o fatti non interpretate, attraverso simboli e contenute su supporti.
- ▶ **Informazione** dal latino *informatio(-nis)* (nel significato di dare forma alla mente, disciplinare, istruire, insegnare). L'informazione è un insieme di dati, interpretati e comprensibili per il destinatario.

Generalità sull'analisi dei dati

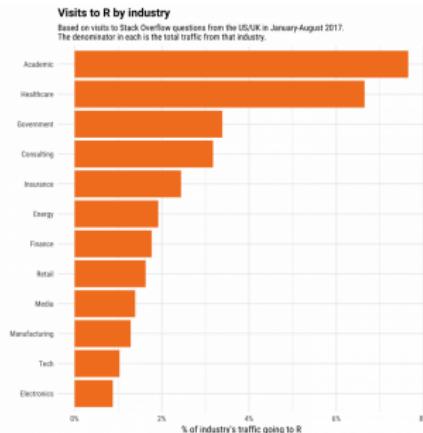
Il flusso delle informazioni



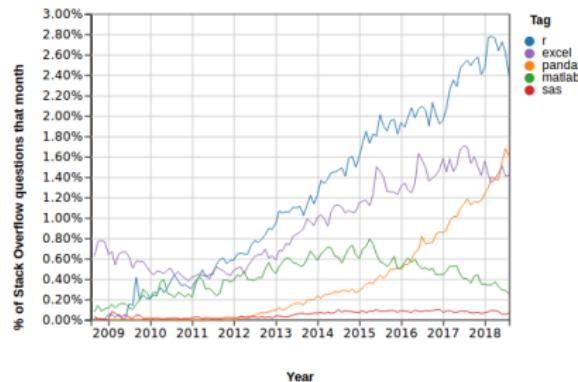
L'ambiente **R** per l'analisi dei dati

L'ambiente **R** per l'analisi dei dati è particolarmente utile in tutti i campi del sapere dove è necessario processare, analizzare e estrarre informazioni da grandi quantità di dati.

Un pò di statistiche (-:



(a)



(b)

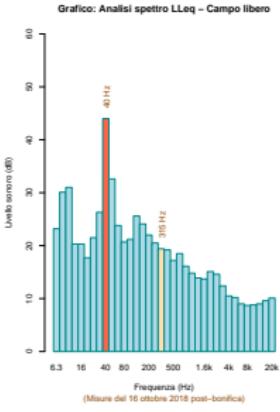
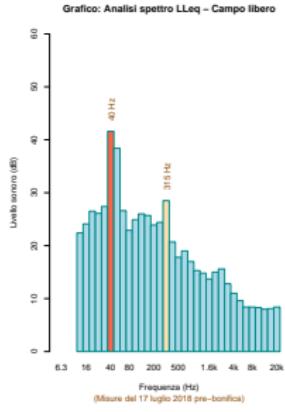
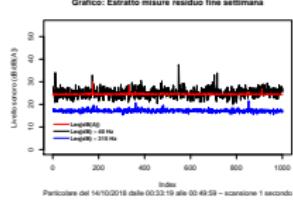
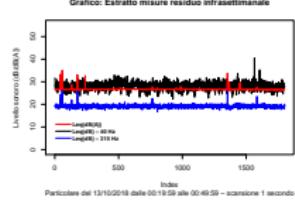
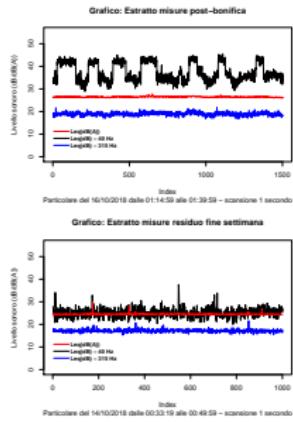
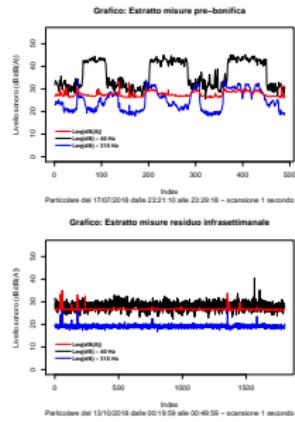
Uso di **R** in ambito acustico ambientale

L'uso di **R** in acustica ambientale diventa indispensabile soprattutto nei monitoraggi acustici di **lungo periodo** e laddove c'è la necessità di **confrontare** diversi dataset (meteo, idrometrici, ecc...) con i dati fonometrici:

- ▶ Reti di monitoraggio **rumore urbano**.
- ▶ Reti di monitoraggio **rumore aeroportuale**.
- ▶ Campagne di misurazione **rumore da traffico stradale e ferroviario**.
- ▶ **Casi particolari di esposto** dove è necessaria una misura prolungata e in continuo per caratterizzare il disturbo della sorgente in esame.

Uso di **R** in ambito acustico ambientale

Esempi elaborazioni grafiche

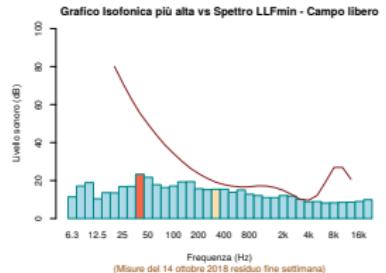
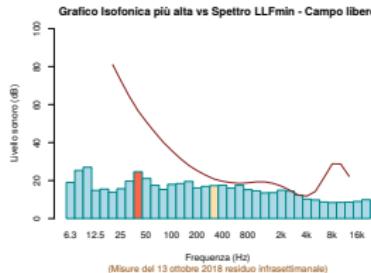
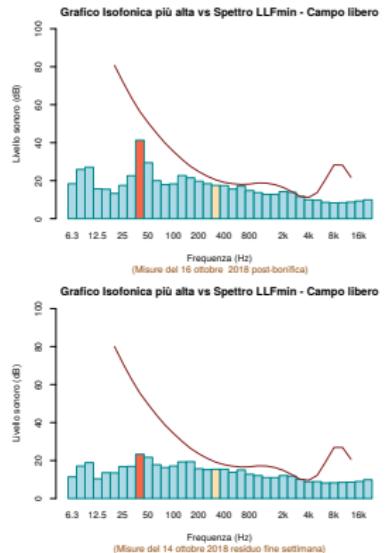
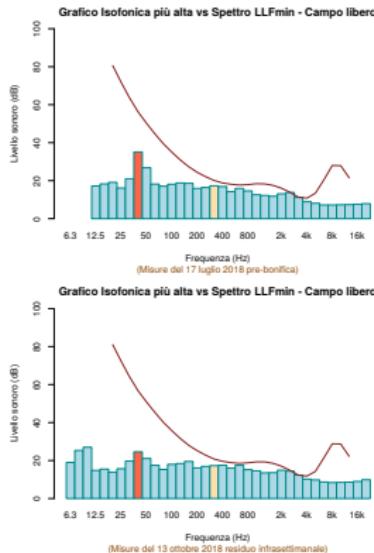


(c)

(d)

Uso di **R** in ambito acustico ambientale

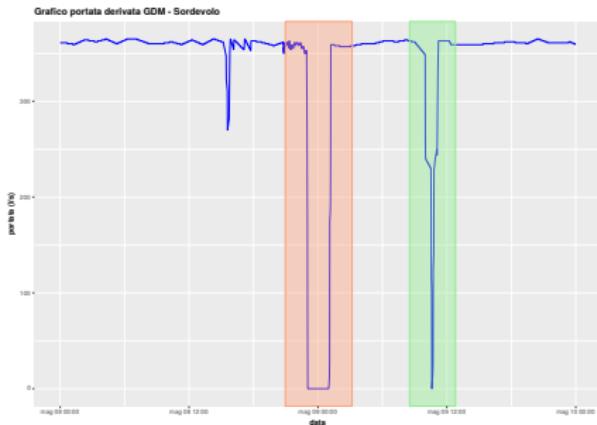
Esempi elaborazioni grafiche



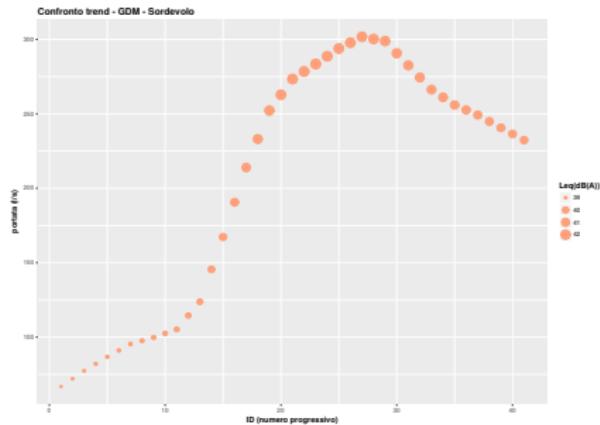
(e)

Uso di **R** in ambito acustico ambientale

Esempi elaborazioni grafiche



(f)



(g)

Uso di **R** in ambito acustico ambientale

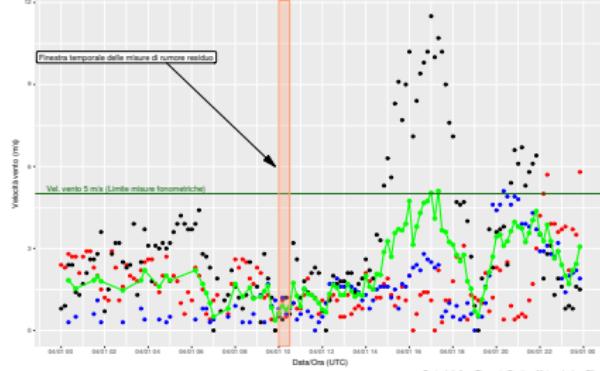
Esempi elaborazioni grafiche

Dati del 22 dicembre 2016 - Staz. meteo ARPA Piemonte
(Piverone/blu, Massazza/rosso, Oropa/nero, Media_Stazioni/verde)



(h)

Dati del 04 gennaio 2017 - Staz. meteo ARPA Piemonte
(Piverone/blu, Massazza/rosso, Oropa/nero, Media_Stazioni/verde)



(i)

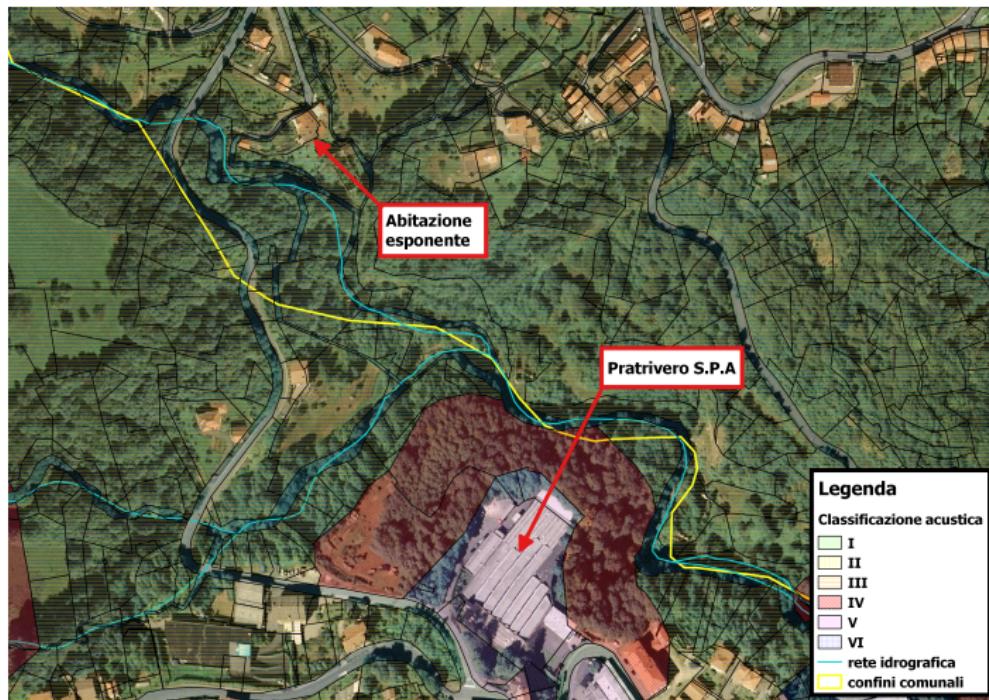
Caso studio: Esposto

L'esponto che esamineremo ricade nei casi complessi in quanto l'esponente lamenta un disturbo rumoroso proveniente da una Azienda che dista dall'abitazione dell'esponente circa 300 metri ed inoltre l'abitazione è costeggiata da un torrente.

- ▶ Circostanze che aumentano la complessità delle misurazioni:
 - **Distanza** fra sorgente e ricettore
 - **Orografia** di tipo vallivo
 - Presenza del **torrente**
 - **Rumore muticiclico modulato**

Per questi motivi per poter caratterizzare al meglio il fenomeno rumoroso è stato necessario effettuare una misurazione in continuo per un periodo prolungato.

Caso studio: Esposto



(j) Foto satellitare del luogo in esame

Caso studio: Esposto

Dopo un **sopralluogo al ricettore** si è provveduto a **pianificare** ed effettuare una serie di interventi di misura sia in esterno che in interno all'abitazione dell'esponente.

Da una prima sommaria analisi delle **misure fonometriche** corredate di **registrazione audio** (indispensabile per le misure in continuo e prolungate) è emerso che il periodo più critico risulava essere quello notturno nella fascia oraria 22:00 - 06:00.

All'interno di questa fascia oraria si sono individuate le ore meno disturbate dall'attività casalinga dell'esponente 00:00 - 05.00.

Le successive analisi pertanto si sono concentrate in questa fascia oraria.

Caso studio: Esposto

I dati raccolti sono stati scaricati dalla strumentazione in formato csv (un file ogni ora di misura). Prima dell'analisi con il software **R** è stato fatto un preliminare ascolto delle tracce audio per identificare univocamente i cicli della sorgente rumorosa.

Quindi **R** è stato utile per:



- ▶ Unire i file orari
- ▶ Marcare tutti i cicli a 40 Hz e 315 Hz sul valore di $Leq(dB(A))$
- ▶ Unire i valori di $Leq(dB(A))$ corrispondenti ai cicli
- ▶ Ricercare eventuali toni puri
- ▶ Calcolare la media energetica oraria sul $Leq(dB(A))$ corrispondente ai cicli
- ▶ Rappresentare graficamente i risultati

Caso studio: Esposto

Nelle seguenti slides analizzeremo passo passo il codice **R** prodotto per il trattamento dei dati.

Codice R

```
1 # Lettura nome file e inserimento nomi in un vettore di stringhe
2 lista.files <- list.files(dirdataset)
3
4 # Carico dataset orari (00:00 – 04:00) dei Giorni 21 – 22 e 23 settembre 2018
5 for (j in 13:17) {
6   assign(paste("df_", substr(lista.files[j], 1, 12), sep = "")),
7   read.table(paste(dirdataset, lista.files[j], sep = "/"),
8   skip = 6, dec = ".", sep = ";", stringsAsFactors = FALSE))
9 }
10
11 for (j in 18:length(lista.files)) {
12   assign(paste("df_", substr(lista.files[j], 1, 13), sep = "")),
13   read.table(paste(dirdataset, lista.files[j], sep = "/"),
14   skip = 6, dec = ".", sep = ";", stringsAsFactors = FALSE))
15 }
```

Caso studio: Esposto

Codice R

```
1 # Operazione comune a tutti i dataset
2 df_210918_00_in.name <- read.table(paste(dirdataset, lista.files[13], sep = "/"), skip = 5,
3 nrows = 1, dec = ".", sep = ";", stringsAsFactors = FALSE)
4
5 # Aggiunta delle lettere "min" dalla colonna 50 alla 87 (spettro dei minimi)
6 df_210918_00_in.name[50:87] <- paste(df_210918_00_in.name[50:87], "min", sep = "")
7 names(df_220918_00_out) <- df_210918_00_in.name
8 # Eliminazione spazi e parentesi nel nome delle variabili
9 names(df_220918_00_out) <- gsub("\\W|\\s", "", names(df_220918_00_out))
10
11 # Inserimento nomi ottimizzati alle variabili
12 names(df_220918_01_out) <- names(df_220918_00_out)
13 names(df_220918_02_out) <- names(df_220918_00_out)
14 names(df_220918_03_out) <- names(df_220918_00_out)
15 names(df_220918_04_out) <- names(df_220918_00_out)
16
17 names(df_230918_00_out) <- names(df_220918_00_out)
18 names(df_230918_01_out) <- names(df_220918_00_out)
19 names(df_230918_02_out) <- names(df_220918_00_out)
20 names(df_230918_03_out) <- names(df_220918_00_out)
21 names(df_230918_04_out) <- names(df_220918_00_out)
```

Caso studio: Esposto

Codice R

```
1 # Funzione personalizzata per creare un sequenza:  
2 # data(YYYY-MM-DD), ora(hh:mm:ss) al secondo  
3 time.sequence <- function(data, ora){  
4   start.seq <- as.POSIXct(paste(data, ora, sep = " "),  
5   format("%Y-%m-%d %H:%m:%s"))  
6   stop.seq <- start.seq + 3599  
7   sequence <- seq(start.seq,  
8   stop.seq, by = 1)  
9   return(sequence)  
10 }  
11  
12 # Ciclo for per creare piu sequenze orarie  
13 # Seguenza del 22 settembre 2018  
14 for (i in 0:4) {  
15   assign(paste0("time2209180", i),  
16   time.sequence("2018-09-22", paste0("0", i, ":00:00")))  
17 }  
18  
19 # Sequenza del 23 settembre 2018  
20 for (i in 0:4) {  
21   assign(paste0("time2309180", i),  
22   time.sequence("2018-09-22", paste0("0", i, ":00:00")))  
23 }
```

Caso studio: Esposto

Codice R

```
1 #Aggiunta della colonna date ricostruita
2 df_220918_00_out$date <- time22091800
3 df_220918_01_out$date <- time22091801
4 df_220918_02_out$date <- time22091802
5 df_220918_03_out$date <- time22091803
6 df_220918_04_out$date <- time22091804
7
8 df_230918_00_out$date <- time23091800
9 df_230918_01_out$date <- time23091801
10 df_230918_02_out$date <- time23091802
11 df_230918_03_out$date <- time23091803
12 df_230918_04_out$date <- time23091804
13
```

Caso studio: Esposto

Codice R

```
1 # Marcatura eventi fonometrici superiori a 40 dB
2 df_220918_00_out$MarkerEU <- as.factor(ifelse(df_220918_00_out$'40Hz' > 42, "M", "N"))
3 df_220918_01_out$MarkerEU <- as.factor(ifelse(df_220918_01_out$'40Hz' > 42, "M", "N"))
4 df_220918_02_out$MarkerEU <- as.factor(ifelse(df_220918_02_out$'40Hz' > 42, "M", "N"))
5 df_220918_03_out$MarkerEU <- as.factor(ifelse(df_220918_03_out$'40Hz' > 42, "M", "N"))
6 df_220918_04_out$MarkerEU <- as.factor(ifelse(df_220918_04_out$'40Hz' > 42, "M", "N"))
7
8 df_230918_00_out$MarkerEU <- as.factor(ifelse(df_230918_00_out$'40Hz' > 42, "M", "N"))
9 df_230918_01_out$MarkerEU <- as.factor(ifelse(df_230918_01_out$'40Hz' > 42, "M", "N"))
10 df_230918_02_out$MarkerEU <- as.factor(ifelse(df_230918_02_out$'40Hz' > 42, "M", "N"))
11 df_230918_03_out$MarkerEU <- as.factor(ifelse(df_230918_03_out$'40Hz' > 42, "M", "N"))
12 df_230918_04_out$MarkerEU <- as.factor(ifelse(df_230918_04_out$'40Hz' > 42, "M", "N"))
```

Caso studio: Esposto

Codice R

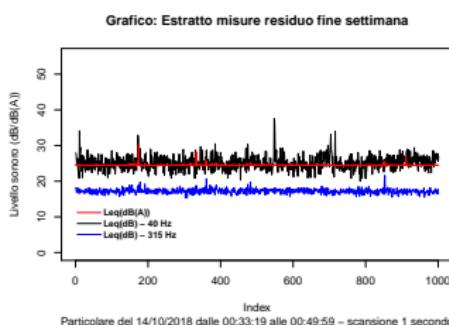
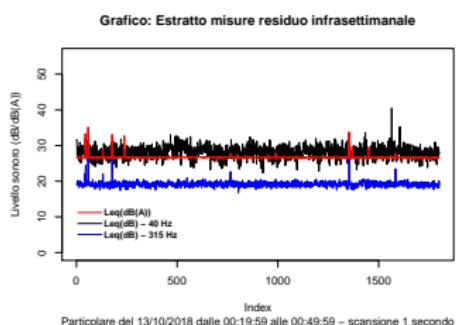
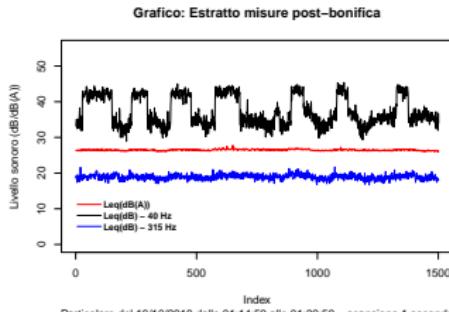
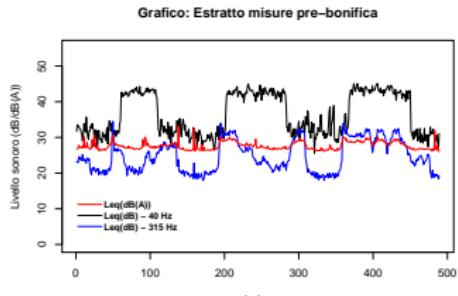
```
1 # Unione dei dataframe
2
3 Unionone_df_2223_09_18 <- rbind(
4   f_220918_00_out,
5   df_220918_01_out,
6   df_220918_02_out,
7   df_220918_03_out,
8   df_220918_04_out,
9
10  df_230918_00_out,
11  df_230918_01_out,
12  df_230918_02_out,
13  df_230918_03_out,
14  df_230918_04_out
15 )
16 )
```

Caso studio: Esposto

Codice R

```
1 # Grafico: Particolare delle misure effettuate
2 plot(df17072018_in$LLeq_40Hz, type = "l", ylim = c(0, 55),
3       ylab = "Livello sonoro (dB/dB(A))",
4       main = "Grafico: Estratto misure pre-bonifica",
5       sub = "Particolare del 17/07/2018 dalle 23:21:10 alle 23:29:18 – scansione 1 secondo")
6 lines(df17072018_in$LLeq_315Hz, col = "blue")
7 lines(df17072018_in$LAeq, col = "red")
8 legend("bottomleft", c("Leq(dB(A))", "Leq(dB) – 40 Hz", "Leq(dB) – 315 Hz"),
9        col = c("red", "black", "blue"), lty = c(1,1,1), inset = 0.01,
10        cex = 0.85, y.intersp = 0.30, x.intersp = 0.2, bty = "n", text.font = 2)
```

Caso studio: Esposto



(k) Analisi esplorativa

Caso studio: Esposto

Codice R

```
1 # Subsetting degli eventi fonometrici superiori a 42 dB
2 Eventi_22_40Hz <- subset(Unione_df_22_09_18, MarkerEU == "M")
3 plot(Eventi_22_40Hz$'40Hz')
4
5 # Isolamento dello spettro dei minimi
6 Eventi_22_40Hz_min <- Eventi_22_40Hz[ , c(1, 3, 42, 50:87)]
7 Eventi_22_40Hz_spettromin <- Eventi_22_40Hz[, c(50:85)]
8
9 # Manipolazione dataframe gathering
10 Eventi_40Hz_spettrominLong <- gather(Eventi_22_40Hz_spettromin, freq, llmin)
11
12 # Creazione di una funzione personalizzata "Calcolo della media energetica
13 # (media logaritmica in base 10)"
14 EnergeticMean <- function(x) {
15   x <- x[!is.na(x)]
16   li <- (1 / length(x))
17   s <- sum(10 ^ (x / 10))
18   return(round(10 * (log10(li * s)), 1))
19 }
```

Caso studio: Esposto

Codice R

```
1 # Calcolo della media energetica
2 Tabella_spettrominimi <- as.data.frame(as.table(
3   tapply(Eventi_40Hz_spettrominLong$llmin ,
4   Eventi_40Hz_spettrominLong$freq , EnergeticMean)))
5
6 # Riordino delle frequenze (ordine perso "da rivedere per ottimizzare")
7 # Questo passo sarebbe stato inutile se all'etichettatura delle colonne avrei
8 # lasciato il punto
9 freq <- c(100, 10, 10000, 12.5, 125, 1250, 12500, 160, 16, 1600, 16000, 1000,
10 200, 20, 20000, 250, 25, 2500, 2000, 31.5, 315, 3150, 400, 40, 4000,
11 500, 50, 5000, 630, 6.3, 63, 6300, 800, 80, 8, 8000)
12
13 Tabella_spettrominimi <- cbind(Tabella_spettrominimi , freq)
14 Tabella_spettrominimi <- Tabella_spettrominimi[order(freq), ]
15 Tabella_spettrominimi <- Tabella_spettrominimi[ , -1]
16 Tabella_spettrominimi <- Tabella_spettrominimi[ , c(2:1)]
17 names(Tabella_spettrominimi) <- c("Hz" , "LLfmin")
18 #-----
```

Caso studio: Esposto

ISO 226 : 1987 (E)

$$L_n = 4.2 + \frac{a_f(L_f - T_f)}{1 + b_f(L_f - T_f)} \quad (1)$$

Codice R

```
1 # Grafico Isofonica vs spettro dei minimi
2 mydata <- Tabella_spettrominimi
3
4 # iso is a data frame of values like table in ISO 226 1987 for free field
5 iso <- read.table("Isofoniche.csv", sep = ";", dec = ".", header = TRUE)
6
7 mydata$LLfmin <- suppressWarnings(as.numeric(as.character(mydata$LLfmin)))
8 isof <- iso
9 isof$LLfmin <- signif(x = mydata$LLfmin, digit = 3)
10 isof$afC <- (isof$af)*(isof$LLfmin - isof$Tf)
11 isof$bfC <- (isof$bf)*(isof$LLfmin - isof$Tf)
12 isof$Ln <- (isof$afC)/(1 + isof$bfC) + 4.2
13 isof$Ln1 <- max(isof$Ln, na.rm = TRUE)
14 isof$A <- (isof$Ln1) - 4.2
15 isof$Lf <- signif(x=(isof$A)/(isof$af-(isof$A)*(isof$bf)) + isof$Tf, digit = 3)
16 isof <- as.data.frame(isof)
```

Caso studio: Esposto

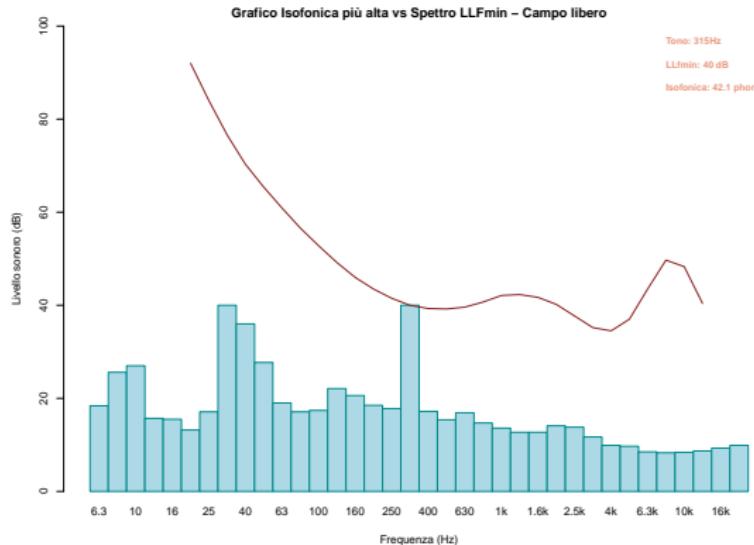
ISO 226 : 1987 (E)

Codice R

```
1 # Codice per la ricerca dei toni puro
2 A <- (length(isof$Hz[(which(isof$LLfmin>=isof$Lf))]) == 1)
3 B <- isof$Hz[(which(isof$LLfmin >= isof$Lf))] == isof$Hz[(which(diff(isof$LLfmin) >= 5) +
4 1))]
5 if (A & A %in% B){
6   paste("Attenzione è presente un tono puro con freq.:",
7   isof$Hz[(which(isof$LLfmin >= isof$Lf))], "Hz", sep = " ")
8 } else {print("Non sono presenti toni puri")}
9
10 # Codice per il grafico spettro dei minimi Vs isofonica più alta
11 mybar <- barplot(isof$LLfmin, names.arg = isof$Hz, col = "blue", ylim = c(0,100),
12 main = "Grafico Isofonica più alta vs Spettro LLFmin – Campo libero",
13 xlab = "Frequenza (Hz)", ylab = "Livello sonoro (dB)", space = 0)
14 lines(mybar, isof$Lf, type = "l", col = "red", lwd = 2)
```

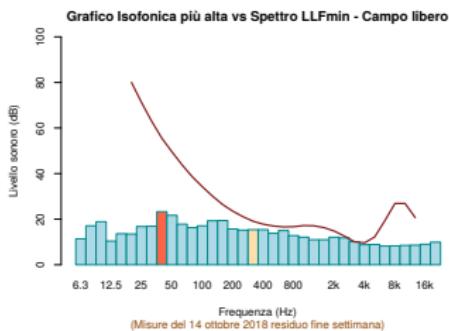
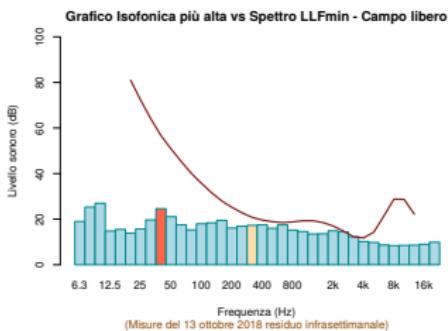
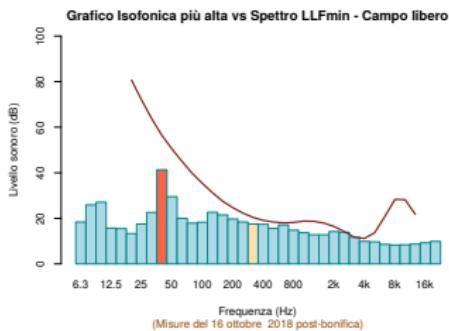
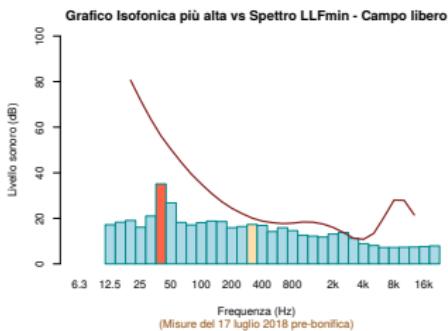
Caso studio: Esposto

Risultato del codice R per la ricerca dei toni



(I) Simulazione

Caso studio: Esposto



(m) Risultati analisi toni puri

Caso studio: Esposto

Grafico: Analisi spettro LLeq – Campo libero

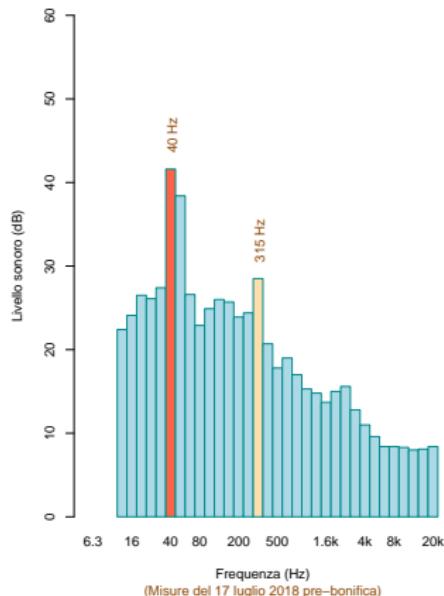
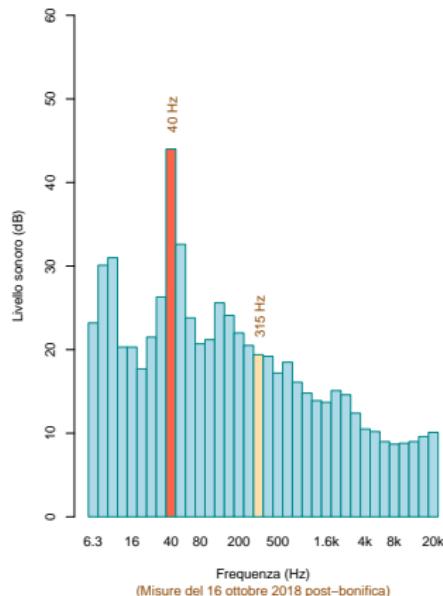


Grafico: Analisi spettro LLeq – Campo libero



(n) Risultati analisi spettrale

Caso studio: Pala Eolica Oulx



In questo caso è stato necessario calcolare la media scalare della velocità del vento nel **periodo diurno** (06:00 - 22.00) e nel **periodo notturno** (22:00 - 06:00) per il confronto con i dati fonometrici.

Caso studio: Pala eolica Oulx

Codice R

```
1 # Carico dataset
2 df <- read.csv(paste(dirdataset, "Vel_VentoSCALARE.csv", sep = "/"),
3                 sep = ";",
4                 dec = ".",
5                 header = T,
6                 stringsAsFactors = F,
7                 na.strings = "null")
8
9 # Manipolazione dataset eliminazione variabili irrilevanti per l'analisi
10 # e cambio nome ad alcune variabili
11 df <- df[, c(9:12)]
12 names(df)[3:4] <- c("VelVento(m/s)", "Validazione")
13
14 # Subsetting dati validati
15 df <- subset(df, Validazione == "OK")
```

Caso studio: Pala eolica Oulx

Codice R

```
1 # Inserisco i flag "giorno" (6–22) e "notte" (22–6)
2 # Manipolazione data e ora
3 df$date <- paste(df$data, df$ora, sep = " ")
4 df$date <- dmy_hm(df$date)
5 df$Ora <- hour(df$date)
6 df$Giorno <- day(df$date)
7 df$Mese <- month(df$date)
8
9 # Inserimento flags nella nuova variabile Periodo
10 df$Periodo1 <- ifelse(df$Ora <= 5, "N", "G")
11 df$Periodo2 <- ifelse(df$Ora >= 22, "N", "G")
12 df$Periodo <-
13 ifelse(df$Periodo1 == "N" | df$Periodo2 == "N", "N", "G")
14 #~~~~~#
15
16 # subsetting df rispetto i flags notte "N" e giorno "G"
17 df_N <- subset(df, Periodo == "N")
18 df_G <- subset(df, Periodo == "G")
19 #~~~~~#
```

Caso studio: Pala eolica Oulx

Codice R

```
1 # Calcolo meadia, min, massimo e deviazione standard periodo Diurno
2 t_G <- tapply(df_G$'VelVento(m/s)', list(df_G$Giorno, df_G$Mese),
3 function(x) {
4   c(
5     MEAN = mean(x),
6     MIN = min(x),
7     MAX = (max(x)),
8     SD = sd(x)
9   )
10 })
11
12 # Assemblo il contenuto della lista "t_N" e lo trasformo in dataframe
13 df_G_Day <- as.data.frame(do.call(rbind, t_G))
14
15 # Inserisco il vettore data nel dataframe
16 df_G_Day$DATA <- unique(df_G$data)
17
18 # Riordino posizione variabili
19 df_G_Day <- df_G_Day[, c(5, 1:4)]
```

Caso studio: Pala eolica Oulx

Codice R

```
1 # Arrotondamenti ad una cifra decimale
2 # Creazione funzione personalizzata per arrotondare
3 round_df <- function(x, decimali) {
4 # arrotonda tutte le variabili numeriche
5 # x: data frame
6 # decimali: numero di decimali
7 numeric_columns <- sapply(x, mode) == 'numeric'
8 x[numeric_columns] <- round(x[numeric_columns], decimali)
9 x
10 }
11
12 # Applico la funzione "round_df"
13 df_G_Day <- round_df(df_G_Day, 2)
14 #~~~~~
15
16 # Salvataggio files .csv elaborato nella cartella "Tabelle_Elaborate"
17 write.table(
18   df_G_Day,
19   paste(
20     dirsalvataggio , "TabGiorno_StatGiornaliereVV_2017_OULX.csv",
21     sep = "/"
22   ),
23   row.names = F,
24   sep = ";",
25   dec = ".")
```

Caso studio: Pala eolica Oulx

Codice R

```
1 # codice per calcolo statistiche notturne a cavallo di due giorni
2 # Inserimento flags alle ore notturne divise per >= 22 "N1" il resto "N2"
3 df_NPeriodoAcu <- ifelse(df_N$Ora >= 22, "N1", "N2")
4 # Eliminazione variabili non necessarie
5 df_N <- df_N[, c(1, 5, 6:8, 3, 4, 11:12)]
6 #~-----
7 #~-----
8 # subsetting per l'eliminazione delle prime e ultime ore che non completano il
9 # range orario da analizzare
10 df_N1 <- df_N[!(df_N$data == unique((df_N$data))[1] & df_N$Ora < 22),]
11
12 df_N2 <- df_N1[!(df_N1$data == unique((df_N1$data))[365] &
13 df_N1$Ora >= 22),]
14 #~-----
15 # ciclo che mette in lista i dataframe subbetti secondo i criteri 22–6 di due
16 # giorni consecutivi
17 a <- list()
18 for (i in 1:length(unique(df_N2$data))) {
19   a[[i]] <- subset(df_N2,
20     Ora %in% c(23, 22, 0, 1, 2, 3, 4, 5) &
21     df_N2$data %in% c(unique((df_N2$data))[i],
22       unique((df_N2$data))[i + 1]))
23   a[[i]] <- a[[i]][!(a[[i]]$data == unique((df_N2$data))[i] &
24     a[[i]]$Ora %in% c(0, 1, 2, 3, 4, 5)),]
25
26   a[[i]] <- a[[i]][!(a[[i]]$data == unique((df_N2$data))[i + 1] &
27     a[[i]]$Ora %in% c(23, 22)),]
28 }
```

Caso studio: Pala eolica Oulx

Codice R

```
1 # funzione personalizzata per il calcolo delle statistiche di base
2 summary_night <- function(x) {
3   c(
4     MEAN = mean(x),
5     MIN = min(x),
6     MAX = max(x),
7     SD = sd(x)
8   )
9 }
10
11 # Applicazione funzione sulla lista contenente 364 dataframe 22-6
12 b <- list()
13 for (j in 1:length(a)) {
14   b[[j]] <- summary_night(a[[j]][6]$'VelVento(m/s)')
15 }
16
17 # Creazione del dataframe finale con i risultati
18 df_nighthACU <- as.data.frame(do.call(rbind, b))
19
20 # Inserimento colonna DATA
21 df_nighthACU$DATA <- unique(df_N2$data)
22
23 # Riarrangiamento colonne del dataframe
24 df_nighthACU <- df_nighthACU[, c(5, 1, 2, 3, 4)]
```

Caso studio: Pala eolica Oulx

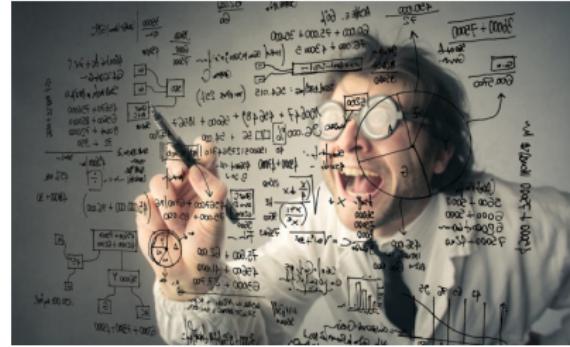
Codice R

```
1 # Arrotondamenti ad n cifre decimali
2 # Creazione funzione personalizzata per arrotondare
3 round_df <- function(x, decimali) {
4 # arrotonda tutte le variabili numeriche
5 # x: data frame
6 # decimali: numero di decimali
7 numeric_columns <- sapply(x, mode) == 'numeric'
8 x[numeric_columns] <- round(x[numeric_columns], decimali)
9 x
10 }
11
12 # Applico la funzione "round_df"
13 df_nighthACU <- round_df(df_nighthACU, 2)
14
15
16 # Salvataggio dataframe in un file csv
17 write.table(
18 df_nighthACU,
19 paste(dirsalvataggio , "TabNotteACU_StatGiornaliereVV_2017_OULX.csv",
20 sep = "/"),
21 sep = ";",
22 dec = ".",
23 row.names = F
24 )
#####
```

Conclusioni

Concludendo si può chiaramente affermare che l'**uso** di **R** nell'ambito dell'**acustica ambientale** e più in generale nell'ambito dell'analisi dei **dati ambientali** sta divenendo **indispensabile** pena la **svalutazione** della **mole importante** di **dati** che ogni giorno, ora, minuto e secondo ARPA produce.

Due risate...



Bibliografia

- ▶ The R manuals - edited by the R Development Core Team.
- ▶ R Core Team (2018). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: <https://www.R-project.org/>.
- ▶ Manuale di acustica applicata - Renato Spagnolo Casa editrice: UTET Edizione: marzo 2004.

Grazie per l'attenzione!

