EyeDOC

Simo

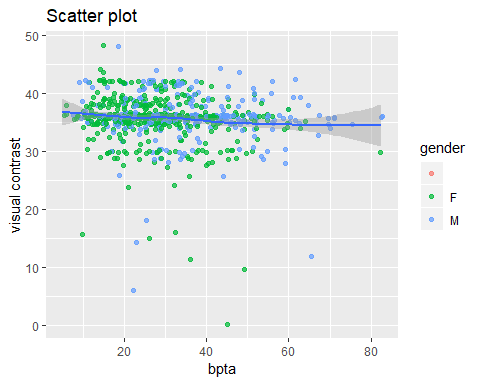
January 31, 2019

# Scatter plot

##1 BPTA

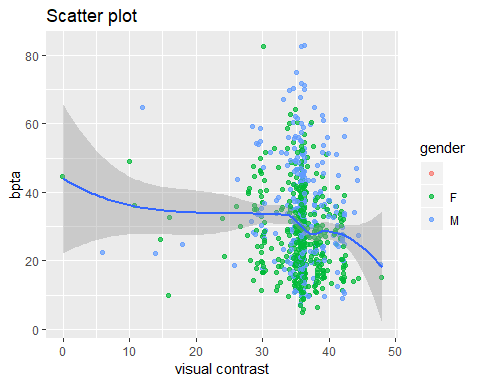
###1.1.1 BPTA & contrast sensitivity

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



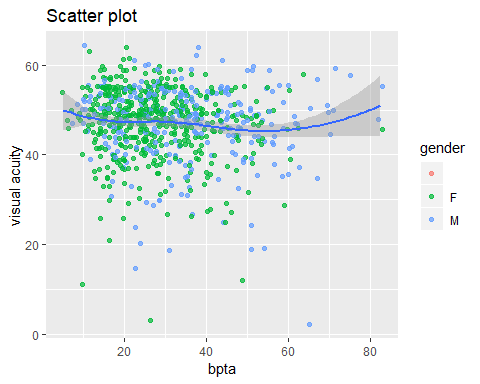
###1.1.2

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'

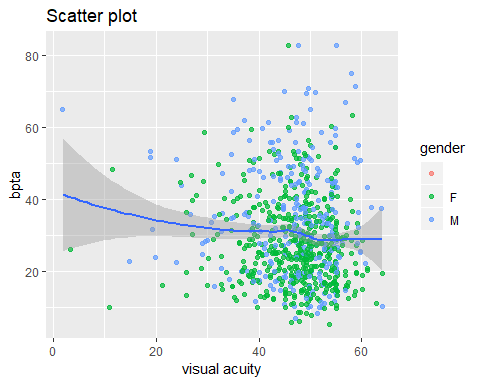


### 1.2 BPTA & Visual Acuity

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



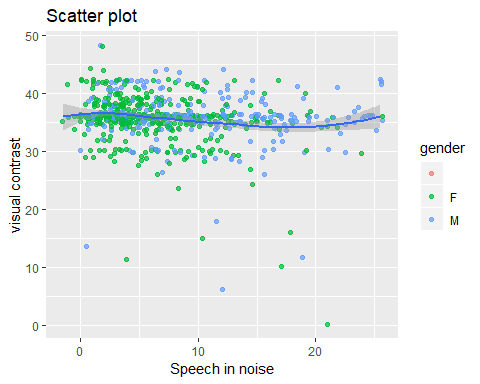
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



## 2 SNR

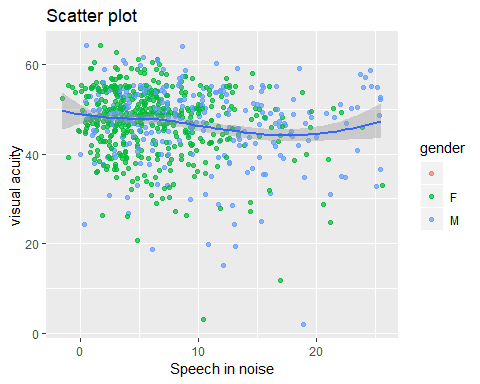
### 2.1 Signal to noise ratio and Visual contrast sensitivity

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



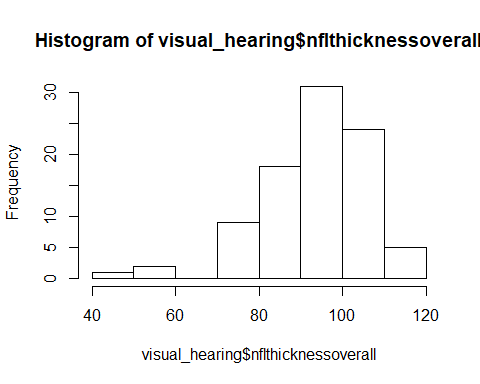
### 2.2 Signal to noise ratio and Visual acuity

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'

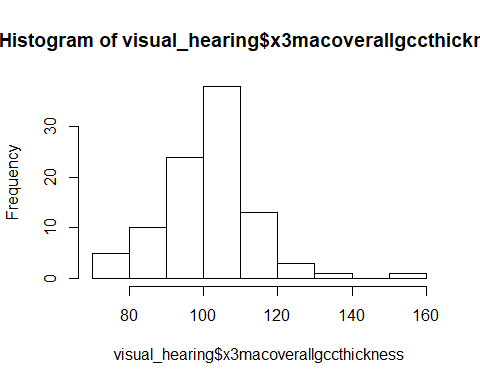
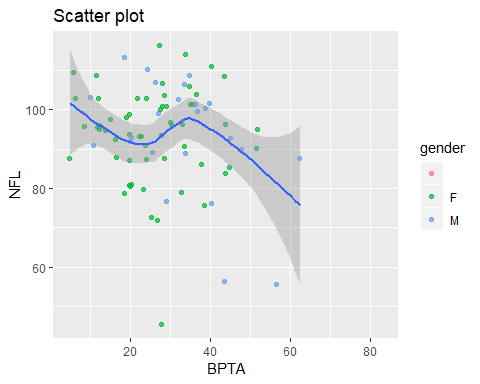


## OCT

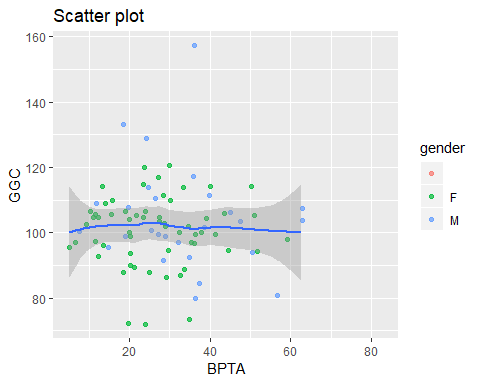
### PTA



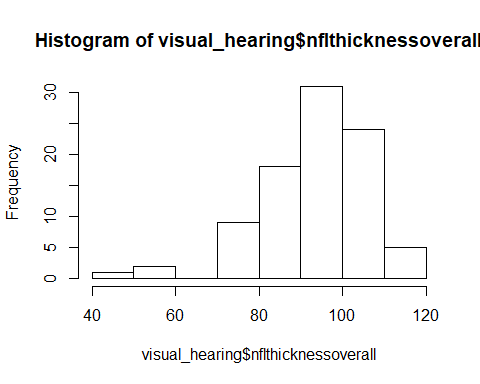
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



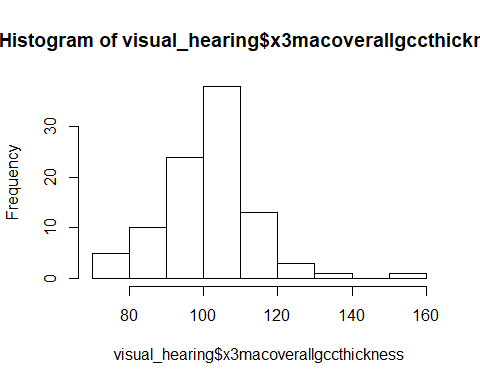
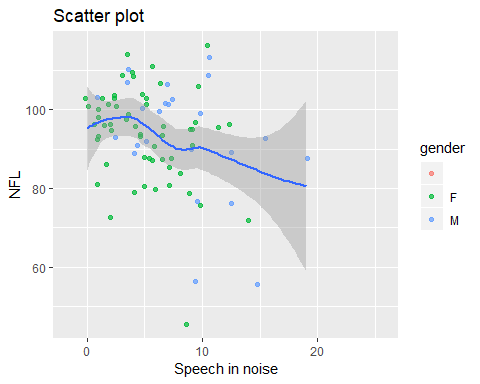
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



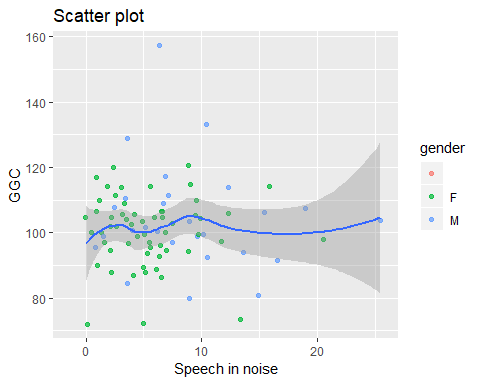
### Speech in noise

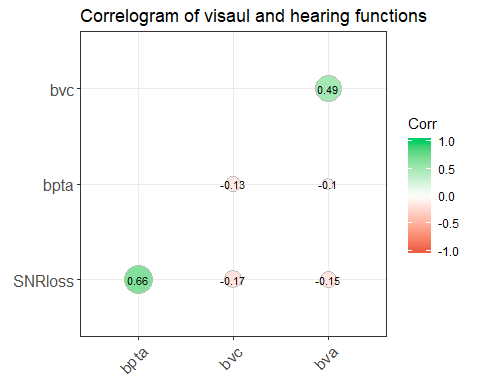


## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'





## function (x, y = NULL, use = "everything", method = c("pearson",   
## "kendall", "spearman"))   
## {  
## na.method <- pmatch(use, c("all.obs", "complete.obs", "pairwise.complete.obs",   
## "everything", "na.or.complete"))  
## if (is.na(na.method))   
## stop("invalid 'use' argument")  
## method <- match.arg(method)  
## if (is.data.frame(y))   
## y <- as.matrix(y)  
## if (is.data.frame(x))   
## x <- as.matrix(x)  
## if (!is.matrix(x) && is.null(y))   
## stop("supply both 'x' and 'y' or a matrix-like 'x'")  
## if (!(is.numeric(x) || is.logical(x)))   
## stop("'x' must be numeric")  
## stopifnot(is.atomic(x))  
## if (!is.null(y)) {  
## if (!(is.numeric(y) || is.logical(y)))   
## stop("'y' must be numeric")  
## stopifnot(is.atomic(y))  
## }  
## Rank <- function(u) {  
## if (length(u) == 0L)   
## u  
## else if (is.matrix(u)) {  
## if (nrow(u) > 1L)   
## apply(u, 2L, rank, na.last = "keep")  
## else row(u)  
## }  
## else rank(u, na.last = "keep")  
## }  
## if (method == "pearson")   
## .Call(C\_cor, x, y, na.method, FALSE)  
## else if (na.method %in% c(2L, 5L)) {  
## if (is.null(y)) {  
## .Call(C\_cor, Rank(na.omit(x)), NULL, na.method, method ==   
## "kendall")  
## }  
## else {  
## nas <- attr(na.omit(cbind(x, y)), "na.action")  
## dropNA <- function(x, nas) {  
## if (length(nas)) {  
## if (is.matrix(x))   
## x[-nas, , drop = FALSE]  
## else x[-nas]  
## }  
## else x  
## }  
## .Call(C\_cor, Rank(dropNA(x, nas)), Rank(dropNA(y,   
## nas)), na.method, method == "kendall")  
## }  
## }  
## else if (na.method != 3L) {  
## x <- Rank(x)  
## if (!is.null(y))   
## y <- Rank(y)  
## .Call(C\_cor, x, y, na.method, method == "kendall")  
## }  
## else {  
## if (is.null(y)) {  
## ncy <- ncx <- ncol(x)  
## if (ncx == 0)   
## stop("'x' is empty")  
## r <- matrix(0, nrow = ncx, ncol = ncy)  
## for (i in seq\_len(ncx)) {  
## for (j in seq\_len(i)) {  
## x2 <- x[, i]  
## y2 <- x[, j]  
## ok <- complete.cases(x2, y2)  
## x2 <- rank(x2[ok])  
## y2 <- rank(y2[ok])  
## r[i, j] <- if (any(ok))   
## .Call(C\_cor, x2, y2, 1L, method == "kendall")  
## else NA  
## }  
## }  
## r <- r + t(r) - diag(diag(r))  
## rownames(r) <- colnames(x)  
## colnames(r) <- colnames(x)  
## r  
## }  
## else {  
## if (length(x) == 0L || length(y) == 0L)   
## stop("both 'x' and 'y' must be non-empty")  
## matrix\_result <- is.matrix(x) || is.matrix(y)  
## if (!is.matrix(x))   
## x <- matrix(x, ncol = 1L)  
## if (!is.matrix(y))   
## y <- matrix(y, ncol = 1L)  
## ncx <- ncol(x)  
## ncy <- ncol(y)  
## r <- matrix(0, nrow = ncx, ncol = ncy)  
## for (i in seq\_len(ncx)) {  
## for (j in seq\_len(ncy)) {  
## x2 <- x[, i]  
## y2 <- y[, j]  
## ok <- complete.cases(x2, y2)  
## x2 <- rank(x2[ok])  
## y2 <- rank(y2[ok])  
## r[i, j] <- if (any(ok))   
## .Call(C\_cor, x2, y2, 1L, method == "kendall")  
## else NA  
## }  
## }  
## rownames(r) <- colnames(x)  
## colnames(r) <- colnames(y)  
## if (matrix\_result)   
## r  
## else drop(r)  
## }  
## }  
## }  
## <bytecode: 0x0000000015fdf570>  
## <environment: namespace:stats>