Figures for Chapter 2

John H Maindonald

October 28, 2012

```
fig2.1 <-
function (form = speed ~ Year, data = subset(cvalues, Year >=
    1862), errors = TRUE, ...)
{
    if (!errors)
        plot(form, data = data, ...)
    else {
        ylim <- with(data, range(c(speed - error, speed + error),
            na.rm = TRUE))
        plot(form, data = data, ylim = ylim, ...)
        with(data, segments(Year, speed - error, Year, speed +
            error))
        with(data, segments(Year - 1.25, speed - error, Year +
            1.25, speed - error))
        with(data, segments(Year - 1.25, speed + error, Year +
            1.25, speed + error))
    }
    obj <- lm(form, data = data)
    abline(obj)
}

fig2.2 <-
function (seed = NULL, N = 10, parset = simpleTheme(pch = 1:N),
    fontsize = list(text = 12, points = 8))
{
    if (!is.null(parset))
        parset$fontsize <- fontsize
    if (!exists("Wages")) {
        library(Ecdat)
        data(Wages)
    }
    if (is.null(Wages$ID))
        Wages$ID <- rep(1:595, each = 7)
    if (!is.null(seed))
        set.seed(seed)
```

```
    chooseN <- sample(1:595, N)
    whichN <- Wages$ID %in% chooseN
    gph <- xyplot(lwage ~ exp, groups = ID, data = Wages, subset = whichN,
        xlab = "Years experience", ylab = "log(Wage)", par.settings = parset,
        type = c("p", "r"))
    gph
}

fig2.3 <-
function (parset = simpleTheme(pch = 16, alpha = 0.8, cex = 1.25),
    fontsize = list(text = 12, points = 8))
{
    if (!is.null(parset))
        parset$fontsize <- fontsize
    library(lattice)
    library(DAAG)
    Site <- with(ant111b, reorder(site, harvwt, FUN = mean))
    gph <- stripplot(Site ~ harvwt, data = ant111b, par.settings = parset,
        xlab = "Harvest weight of corn")
    gph
}

fig2.4 <-
function (parset = simpleTheme(pch = c(0, 1), cex = 1.2), fontsize = list(text = 12,
    points = 8), annotate = TRUE)
{
    if (!is.null(parset))
        parset$fontsize <- fontsize
    gph <- xyplot(Time ~ Distance, groups = roadORtrack, data = worldRecords,
        scales = list(log = 10, tck = -0.4, x = list(at = 10^c((-1):2)),
            y = list(at = 10^(0:3))))
    gph <- update(gph, xlab = "Distance (s, km)", ylab = "Time (t, min)",
        par.settings = parset, auto.key = list(columns = 2))
    gph1 <- xyplot(Time ~ Distance, data = worldRecords, scales = list(log = 10),
        type = "r")
    gph2 <- gph + as.layer(gph1)
    if (annotate) {
        layer3 <- layer(longd <- log10(290.2), longt <- log10(24 *
            60), panel.arrows(-1, -0.02, -1, -0.64, length = 0.1,
            col = "gray45"), panel.text(-1 + 0.125, -0.06, "100m",
            pos = 3, cex = 1.05, col = "gray45"), panel.arrows(longd,
            longt + 0.7, longd, longt + 0.15, length = 0.1, col = "gray45"),
            panel.text(longd + 0.18, longt + 0.65, "290km", pos = 3,
                cex = 1.05, col = "gray45"), panel.arrows(-1 -
                0.5, -0.79, -1 - 0.12, -0.79, length = 0.1, col = "gray45"),
            panel.text(-1 - 0.47, -0.79, "9.6sec", pos = 2, cex = 1.05,
```

```
                          col = "gray45"), panel.arrows(longd - 0.5, longt,
                          longd - 0.12, longt, length = 0.1, col = "gray45"),
                    panel.text(longd - 0.48, longt, "24h", pos = 2, cex = 1.05,
                          col = "gray45"))
              gph2 <- gph2 + layer3
         }
         gph2
    }

    fig2.5 <-
    function (parset = simpleTheme(lty = c(2, 1, 2), col.line = c("gray30",
         "black", "gray30"), pch = c(0, 1)), fontsize = list(text = 12,
         points = 8))
    {
         if (!is.null(parset))
              parset$fontsize <- fontsize
         wr.lm <- lm(log(Time) ~ log(Distance), data = worldRecords)
         resid1 <- resid(wr.lm)
         library(mgcv)
         wr.gam <- gam(resid1 ~ s(log(Distance)), data = worldRecords)
         hat.gam <- predict(wr.gam, se.fit = TRUE)
         wrgamdata <- with(worldRecords, data.frame(distance = Distance,
              roadORtrack = roadORtrack, resid1 = resid1, resid2 = resid(wr.gam),
              hat = hat.gam$fit, se = hat.gam$se.fit))
         ord <- with(wrgamdata, order(distance))
         wrgamdata <- wrgamdata[ord, ]
         library(latticeExtra)
         gph0 <- xyplot(resid1 ~ distance, groups = roadORtrack, ylim = c(-0.15,
              0.175), xlab = "", scales = list(x = list(log = 10, alternating = 0),
              tck = -0.4), data = wrgamdata, type = "p", par.settings = parset,
              auto.key = list(columns = 2))
         gph1 <- xyplot(I(hat - 2 * se) + hat + I(hat + 2 * se) ~
              distance, outer = FALSE, ylim = c(-0.125, 0.175), scales = list(tck = -0.4,
              x = list(log = 10, alternating = 2)), data = wrgamdata,
              type = "l", par.settings = parset)
         gph01 <- update(gph0 + as.layer(gph1), ylab = expression(atop(Smooth %+-%
              2 * SE, "(resid1)")))
         gph2 <- xyplot(resid2 ~ distance, groups = roadORtrack, scales = list(tck = -0.4,
              x = list(log = 10)), ylim = c(-0.125, 0.175), ylab = expression(atop("Residuals fro
              "(resid2)")), data = wrgamdata, type = c("p"), par.settings = parset)
         list(upper = gph01 + as.layer(gph0, axes = NULL), lower = gph2)
    }

    fig2.6 <-
    function (data = loti)
    {
```

```
    anom <- data[, "J.D"]
    num <- seq(along = anom)
    AVtodate <- cumsum(anom)/num
    yr <- data$Year
    plot(anom ~ yr, xlab = "Year", ylab = expression("Difference from 1951-1980 (" *
        degree * "C)"))
    lines(AVtodate ~ yr, col = "gray", lwd = 2)
    lastLessYr <- max(yr[anom < AVtodate])
    lastLessy <- data[as.character(lastLessYr), "J.D"]
    yarrow <- lastLessy - c(4, 0.75) * strheight("O")
    arrows(lastLessYr, yarrow[1], lastLessYr, yarrow[2], col = "gray",
        lwd = 2)
    title1 <- expression("Annual global temperature anomalies, in 0.01" *
        degree * "C,")
    title(main = title1, line = 2.1)
    title2 <- expression("from the average (" %~~% 14 * degree *
        "C) between 1951 and 1980")
    title(main = title2, line = 0.8)
}

fig2.7 <-
function (statistics = c("airbagAvail", "airbagDeploy", "Restraint"),
    restrict = "!is.na(age)&age>=16&age<998")
{
    library(lattice)
    gph <- plotFars(data = FARS, restrict = restrict)
    plotchars <- c(1:length(statistics))
    plotchars[1] <- 16
    gph <- update(gph, xlab = "", ylab = "Death rate ratio of ratios, w/wo",
        scales = list(tck = 0.5), par.settings = simpleTheme(pch = plotchars))
    gph
}
```