PINP\_Bird\_detection

Natasha Hardy

December 2, 2019

## PINP Bird Detections

This R markdown includes data analyses and visualisations for the bird detections project. Predator species: long-nosed fur seals; samples: scats; Locations: Cape Bridgewater, Montague Island, Gabo Island and Lady Julia Percy Sample collection dates: 2015-2017

Project objective: to compare the detection rates of birds in long-nose fur seal scats across the range of this predator in SE Australia from Bass Strait to southern NSW, and over multiple seasons.

#checking that my setup chunk works, it's been a PIA lately, may need to update to latest versions of R and RStudio...lame  
  
getwd()

## [1] "/Users/natasha/Documents/Professional/WORK/CURRENT JOBS/PINP/PINP DATA/pinp\_stats/data"

# Packages  
  
#Load  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────────────────────────────────────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.2.1 ✔ purrr 0.3.2  
## ✔ tibble 2.1.3 ✔ dplyr 0.8.3  
## ✔ tidyr 1.0.0 ✔ stringr 1.4.0  
## ✔ readr 1.3.1 ✔ forcats 0.4.0

## ── Conflicts ────────────────────────────────────────────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(tidyr)  
#library(reshape)  
library(reshape2)

##   
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':  
##   
## smiths

library(plyr)

## -------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## -------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

## The following object is masked from 'package:purrr':  
##   
## compact

library(dplyr)  
library(PNWColors)  
  
#Note that loading all these packages is hell because of overlap in functions, some packages being superceded but still requiring each other or requiring functions from one package to another. You'll notice I had to force a lot of functions by calling the package as follows   
#data %>% package::function(etc)

# Bird detections data from four export pivot table tabs as .csv  
#https://docs.google.com/spreadsheets/d/1D6EqzmLSjwK2ENXtqOLQHwNu65CnmqUDlcJg-lPaZMI/edit#gid=1310896506  
#Sometimes Rmd plays up with the setup chunk and you may need to force entry to the data using:  
#read.csv("/Users/natasha/Documents/Professional/WORK/CURRENT JOBS/PINP/PINP DATA/pinp\_stats/data/bird\_detection.csv", header=TRUE)  
  
#1 = "bird\_detect\_med" tab  
birdD\_wide\_med <- read.csv("/Users/natasha/Documents/Professional/WORK/CURRENT JOBS/PINP/PINP DATA/pinp\_stats/data/bird\_detect\_med.csv", header=TRUE)  
str(birdD\_wide\_med)

## 'data.frame': 99 obs. of 10 variables:  
## $ sample\_id : Factor w/ 99 levels "CBW-1/16-1","CBW-1/16-10",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ location\_id : Factor w/ 4 levels "CB","GI","LJP",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ location : Factor w/ 4 levels "Cape Bridgewater",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ time\_id : Factor w/ 5 levels "Jan\_16","Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ group\_id : Factor w/ 8 levels "CB\_Jan\_16","CB\_Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ SUM.of.allbirds\_HP : int 0 1 1 1 0 0 1 0 0 1 ...  
## $ SUM.of.dna\_stringency\_med: int 0 0 0 1 1 0 1 0 0 0 ...  
## $ SUM.of.dna\_or\_hp : int 0 1 1 1 1 0 1 0 0 1 ...  
## $ SUM.of.both\_bird : int 0 0 0 1 0 0 1 0 0 0 ...  
## $ SUM.of.both\_sametaxa : int 0 0 0 0 0 0 1 0 0 0 ...

#2 = "bird\_detect\_high" tab  
birdD\_wide\_high <- read.csv("/Users/natasha/Documents/Professional/WORK/CURRENT JOBS/PINP/PINP DATA/pinp\_stats/data/bird\_detect\_high.csv", header=TRUE)  
  
#3 = "penguin\_detect\_med" tab  
penguin\_wide\_med <- read.csv("/Users/natasha/Documents/Professional/WORK/CURRENT JOBS/PINP/PINP DATA/pinp\_stats/data/penguin\_detect\_med.csv", header=TRUE)  
  
#4 = "penguin\_detect\_high" tab  
penguin\_wide\_high <- read.csv("/Users/natasha/Documents/Professional/WORK/CURRENT JOBS/PINP/PINP DATA/pinp\_stats/data/penguin\_detect\_high.csv", header=TRUE)

## Data manipulations

I chose to input data as columns in the master document tab “BIRD\_DATA” in PINP\_BIRD\_DATA.xlsx on GoogleDrive. This is because many columns were calculated based off other columns (such as medium stringency DNA OTU total abundance used to calculate a 1% cut-off for high stringency DNA OTU columns). This means a little data wrangling on this end to merge these data into long-form for analyses and graphing.

For the following manipulations, I’m going to: (i) Rename colnames for each of the four datasets exported, (ii) Transform from wide to long data, using melt(), and include arguments for renaming the output variable and value to “metric” and “detection”, and (iii) Add a categorical descriptor for each method, to later merge datasets 1+2 (bird detections) and 3+4 (penguin detections)

Columns are renamed as follows: hp = hard-part analysis to ID prey dna = DNA metabarcoding to ID prey dna\_or\_hp = detections using either method both\_methods = both detected birds/penguins same\_taxon = both detected the same bird taxon (for bird only, as “both” and “same” are equal for looking at one taxon such as penguins)

Then add descriptor: medium = standard QC for DNA based analysis high = additional exclusion of trace amounts of DNA that passed QC

## 1 Bird detections using medium stringency DNA QC + hard-parts  
#Piping multiple steps  
  
#str(birdD\_wide\_med)  
#colnames(birdD\_wide\_med)  
  
birdD\_med = birdD\_wide\_med %>%   
 dplyr::rename(hp = `SUM.of.allbirds\_HP`, dna = `SUM.of.dna\_stringency\_med`,  
 dna\_or\_hp = `SUM.of.dna\_or\_hp`, both\_methods = `SUM.of.both\_bird`,   
 same\_taxon = `SUM.of.both\_sametaxa`) %>%  
 #Renamed columns  
 reshape2::melt(id.vars=c("sample\_id", "location\_id", "location", "time\_id", "group\_id"), variable.name = "metric", value.name = "detection") %>%  
 #Transform wide to long data, using melt(), and added arguments for renaming the output variable and value to "metric" and "detection"  
 mutate(dna\_qc = "medium")  
 #Added the character vector/factor for merging with the high stringency data  
  
#Looking good:  
str(birdD\_med)

## 'data.frame': 495 obs. of 8 variables:  
## $ sample\_id : Factor w/ 99 levels "CBW-1/16-1","CBW-1/16-10",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ location\_id: Factor w/ 4 levels "CB","GI","LJP",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ location : Factor w/ 4 levels "Cape Bridgewater",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ time\_id : Factor w/ 5 levels "Jan\_16","Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ group\_id : Factor w/ 8 levels "CB\_Jan\_16","CB\_Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ metric : Factor w/ 5 levels "hp","dna","dna\_or\_hp",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ detection : int 0 1 1 1 0 0 1 0 0 1 ...  
## $ dna\_qc : chr "medium" "medium" "medium" "medium" ...

## 2 Bird detections using high stringency DNA QC + hard-parts  
#Piping multiple steps  
  
#colnames(birdD\_wide\_high)  
#need to check and change for column names from this document  
  
birdD\_high = birdD\_wide\_high %>%   
 dplyr::rename(hp = `SUM.of.allbirds\_hp`, dna = `SUM.of.allbirds\_dna\_high`,  
 dna\_or\_hp = `SUM.of.allbirds\_high\_dna\_or\_hp`,   
 both\_methods = `SUM.of.allbirds\_high\_both`,   
 same\_taxon = `SUM.of.allbirds\_high\_same`) %>%  
 #Renamed columns  
 reshape2::melt(id.vars=c("sample\_id", "location\_id", "location", "time\_id", "group\_id"), variable.name = "metric", value.name = "detection") %>%  
 #Transform wide to long data, using melt(), and added arguments for renaming the output variable and value to "metric" and "detection"  
 mutate(dna\_qc = "high")  
 #Added the character vector/factor for merging with the high stringency data  
  
#Looking good:  
str(birdD\_high)

## 'data.frame': 495 obs. of 8 variables:  
## $ sample\_id : Factor w/ 99 levels "CBW-1/16-1","CBW-1/16-10",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ location\_id: Factor w/ 4 levels "CB","GI","LJP",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ location : Factor w/ 4 levels "Cape Bridgewater",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ time\_id : Factor w/ 5 levels "Jan\_16","Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ group\_id : Factor w/ 8 levels "CB\_Jan\_16","CB\_Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ metric : Factor w/ 5 levels "hp","dna","dna\_or\_hp",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ detection : int 0 1 1 1 0 0 1 0 0 1 ...  
## $ dna\_qc : chr "high" "high" "high" "high" ...

## 2 Bird detections using high stringency DNA QC + hard-parts  
#Piping multiple steps  
  
#colnames(penguin\_wide\_med)  
#need to check and change for column names from this document  
  
penguin\_med = penguin\_wide\_med %>%   
 dplyr::rename(hp = `SUM.of.penguin\_hp`, dna = `SUM.of.penguin\_dna\_med`,  
 dna\_or\_hp = `SUM.of.penguin\_med\_dna\_or\_hp`,   
 same\_taxon = `SUM.of.penguin\_med\_both`) %>%  
 #Renamed columns  
 reshape2::melt(id.vars=c("sample\_id", "location\_id", "location", "time\_id", "group\_id"), variable.name = "metric", value.name = "detection") %>%  
 #Transform wide to long data, using melt(), and added arguments for renaming the output variable and value to "metric" and "detection"  
 mutate(dna\_qc = "medium")  
 #Added the character vector/factor for merging with the high stringency data  
  
#Looking good:  
str(penguin\_med)

## 'data.frame': 396 obs. of 8 variables:  
## $ sample\_id : Factor w/ 99 levels "CBW-1/16-1","CBW-1/16-10",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ location\_id: Factor w/ 4 levels "CB","GI","LJP",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ location : Factor w/ 4 levels "Cape Bridgewater",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ time\_id : Factor w/ 5 levels "Jan\_16","Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ group\_id : Factor w/ 8 levels "CB\_Jan\_16","CB\_Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ metric : Factor w/ 4 levels "hp","dna","dna\_or\_hp",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ detection : int 0 1 1 1 0 0 1 0 0 0 ...  
## $ dna\_qc : chr "medium" "medium" "medium" "medium" ...

## 2 Bird detections using high stringency DNA QC + hard-parts  
#Piping multiple steps  
  
#colnames(penguin\_wide\_high)  
#need to check and change for column names from this document  
  
penguin\_high = penguin\_wide\_high %>%   
 dplyr::rename(hp = `SUM.of.penguin\_hp`, dna = `SUM.of.penguin\_dna\_high`,  
 dna\_or\_hp = `SUM.of.penguin\_high\_dna\_or\_hp`,   
 same\_taxon = `SUM.of.penguin\_high\_both`) %>%  
 #Renamed columns  
 reshape2::melt(id.vars=c("sample\_id", "location\_id", "location", "time\_id", "group\_id"), variable.name = "metric", value.name = "detection") %>%  
 #Transform wide to long data, using melt(), and added arguments for renaming the output variable and value to "metric" and "detection"  
 mutate(dna\_qc = "high")  
 #Added the character vector/factor for merging with the high stringency data  
  
#Looking good:  
str(penguin\_high)

## 'data.frame': 396 obs. of 8 variables:  
## $ sample\_id : Factor w/ 99 levels "CBW-1/16-1","CBW-1/16-10",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ location\_id: Factor w/ 4 levels "CB","GI","LJP",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ location : Factor w/ 4 levels "Cape Bridgewater",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ time\_id : Factor w/ 5 levels "Jan\_16","Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ group\_id : Factor w/ 8 levels "CB\_Jan\_16","CB\_Jan\_17",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ metric : Factor w/ 4 levels "hp","dna","dna\_or\_hp",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ detection : int 0 1 1 1 0 0 1 0 0 0 ...  
## $ dna\_qc : chr "high" "high" "high" "high" ...

#column or rowbind?

#Overall bird detection by method: need sum of all samples containing bird by method of detection  
#Need to scale to 100%  
  
#1 Bird medium stringency  
bird\_sum\_med <- ddply(birdD\_med, "metric", summarise,  
 `Detected` = sum(detection, na.rm=TRUE)\*100/99,  
 `Not detected` = (99 - sum(detection, na.rm=TRUE)\*100/99))  
str(bird\_sum\_med)

## 'data.frame': 5 obs. of 3 variables:  
## $ metric : Factor w/ 5 levels "hp","dna","dna\_or\_hp",..: 1 2 3 4 5  
## $ Detected : num 29.29 29.29 45.45 13.13 9.09  
## $ Not detected: num 69.7 69.7 53.5 85.9 89.9

bird\_detect\_med <- melt(bird\_sum\_med, id.vars="metric", variable.name = "detection", value.name = "percentage")  
str(bird\_detect\_med)

## 'data.frame': 10 obs. of 3 variables:  
## $ metric : Factor w/ 5 levels "hp","dna","dna\_or\_hp",..: 1 2 3 4 5 1 2 3 4 5  
## $ detection : Factor w/ 2 levels "Detected","Not detected": 1 1 1 1 1 2 2 2 2 2  
## $ percentage: num 29.29 29.29 45.45 13.13 9.09 ...

#2 Bird high stringency  
bird\_sum\_high <- ddply(birdD\_high, "metric", summarise,  
 `Detected` = sum(detection, na.rm=TRUE)\*100/99,  
 `Not detected` = (99 - sum(detection, na.rm=TRUE)\*100/99))  
str(bird\_sum\_high)

## 'data.frame': 5 obs. of 3 variables:  
## $ metric : Factor w/ 5 levels "hp","dna","dna\_or\_hp",..: 1 2 3 4 5  
## $ Detected : num 29.29 15.15 37.37 7.07 4.04  
## $ Not detected: num 69.7 83.8 61.6 91.9 95

bird\_detect\_high <- melt(bird\_sum\_high, id.vars="metric", variable.name = "detection", value.name = "percentage")  
str(bird\_detect\_high)

## 'data.frame': 10 obs. of 3 variables:  
## $ metric : Factor w/ 5 levels "hp","dna","dna\_or\_hp",..: 1 2 3 4 5 1 2 3 4 5  
## $ detection : Factor w/ 2 levels "Detected","Not detected": 1 1 1 1 1 2 2 2 2 2  
## $ percentage: num 29.29 15.15 37.37 7.07 4.04 ...

#3 Penguin medium stringency  
penguin\_sum\_med <- ddply(penguin\_med, "metric", summarise,  
 `Detected` = sum(detection, na.rm=TRUE)\*100/99,  
 `Not detected` = (99 - sum(detection, na.rm=TRUE)\*100/99))  
str(penguin\_sum\_med)

## 'data.frame': 4 obs. of 3 variables:  
## $ metric : Factor w/ 4 levels "hp","dna","dna\_or\_hp",..: 1 2 3 4  
## $ Detected : num 25.3 17.2 34.3 13.1  
## $ Not detected: num 73.7 81.8 64.7 85.9

penguin\_detect\_med <- melt(penguin\_sum\_med, id.vars="metric", variable.name = "detection", value.name = "percentage")  
str(bird\_detect\_med)

## 'data.frame': 10 obs. of 3 variables:  
## $ metric : Factor w/ 5 levels "hp","dna","dna\_or\_hp",..: 1 2 3 4 5 1 2 3 4 5  
## $ detection : Factor w/ 2 levels "Detected","Not detected": 1 1 1 1 1 2 2 2 2 2  
## $ percentage: num 29.29 29.29 45.45 13.13 9.09 ...

#4 Penguin high stringency  
penguin\_sum\_high <- ddply(penguin\_high, "metric", summarise,  
 `Detected` = sum(detection, na.rm=TRUE)\*100/99,  
 `Not detected` = (99 - sum(detection, na.rm=TRUE)\*100/99))  
str(penguin\_sum\_high)

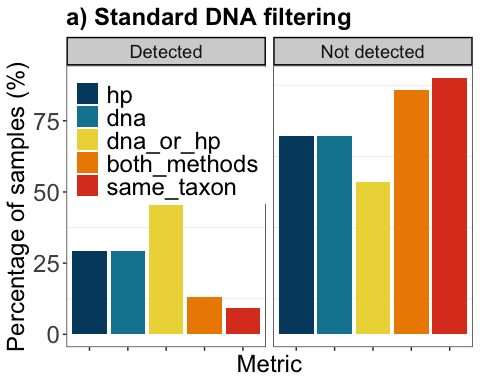
## 'data.frame': 4 obs. of 3 variables:  
## $ metric : Factor w/ 4 levels "hp","dna","dna\_or\_hp",..: 1 2 3 4  
## $ Detected : num 25.25 3.03 26.26 3.03  
## $ Not detected: num 73.7 96 72.7 96

penguin\_detect\_high <- melt(penguin\_sum\_high, id.vars="metric", variable.name = "detection", value.name = "percentage")  
str(bird\_detect\_high)

## 'data.frame': 10 obs. of 3 variables:  
## $ metric : Factor w/ 5 levels "hp","dna","dna\_or\_hp",..: 1 2 3 4 5 1 2 3 4 5  
## $ detection : Factor w/ 2 levels "Detected","Not detected": 1 1 1 1 1 2 2 2 2 2  
## $ percentage: num 29.29 15.15 37.37 7.07 4.04 ...

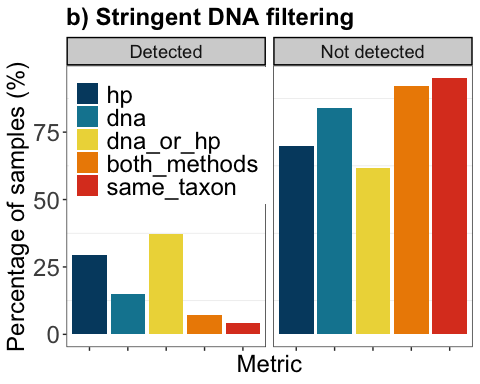
#Can double check this with table()  
#table(birdD\_long$metric, birdD\_long$value)  
  
##Note on renaming values in rows as groups of values:  
#d$Var2[d$Var1 == "C"] <- "Medium"

#Bird detections medium stringency DNA QC  
  
#Colour palette  
fig2.pal <- pnw\_palette(5, name = "Bay", type = "continuous")  
  
#Figure 2  
bird\_fig2a <- ggplot(bird\_detect\_med, aes(x=metric, y=percentage, fill=metric)) +  
 geom\_bar(stat="identity",position="dodge")+  
 #ggtitle("a) Standard DNA filtering")+  
 theme\_bw() + theme(panel.grid.major = element\_blank()) +  
 theme(plot.title=element\_text(face="bold", size=18))+  
 theme(axis.title=element\_text(size=18)) + theme(axis.text=element\_text(size=18)) +  
 theme(axis.text.x=element\_blank())+  
 theme(legend.text=element\_text(size=18)) + #theme(legend.title=element\_text(size=14)) +  
 theme(legend.position=c(0.2,0.75), legend.justification=c(0.4,0.5))+  
 scale\_fill\_manual(values=fig2.pal)+  
 guides(fill=guide\_legend(title=NULL))+  
 labs(title ="a) Standard DNA filtering", x="Metric", y="Percentage of samples (%)") +  
 facet\_grid(.~detection)+  
 theme(strip.text = element\_text(size=rel(1.25)), #face="bold",   
 strip.background = element\_rect(fill="lightgrey", colour="black",  
 size=1))  
bird\_fig2a



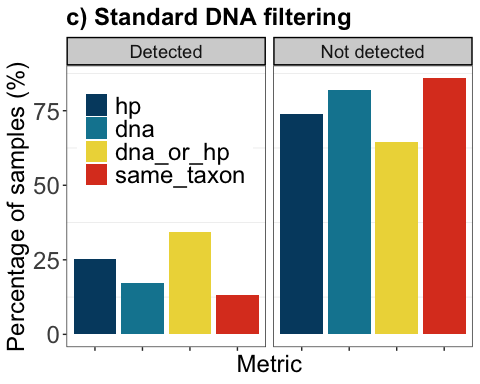
ggsave('figure2a\_birddetectmed.jpeg', plot = bird\_fig2a, width = 7.5, height = 5, dpi = 300)

#Bird detections high stringency DNA QC  
  
#Colour palette  
fig2.pal <- pnw\_palette(5, name = "Bay", type = "continuous")  
  
#Figure 2  
bird\_fig2b <- ggplot(bird\_detect\_high, aes(x=metric, y=percentage, fill=metric)) +  
 geom\_bar(stat="identity",position="dodge")+  
 theme\_bw() + theme(panel.grid.major = element\_blank()) +  
 theme(plot.title=element\_text(face="bold", size=18))+  
 theme(axis.title=element\_text(size=18)) + theme(axis.text=element\_text(size=18)) +  
 theme(axis.text.x=element\_blank())+  
 theme(legend.text=element\_text(size=18)) + #theme(legend.title=element\_text(size=14)) +  
 theme(legend.position=c(0.2,0.75), legend.justification=c(0.4,0.5))+  
 scale\_fill\_manual(values=fig2.pal)+  
 guides(fill=guide\_legend(title=NULL))+  
 labs(title ="b) Stringent DNA filtering", x="Metric", y="Percentage of samples (%)") +  
 facet\_grid(.~detection)+  
 theme(strip.text = element\_text(size=rel(1.25)), #face="bold",   
 strip.background = element\_rect(fill="lightgrey", colour="black",  
 size=1))  
bird\_fig2b



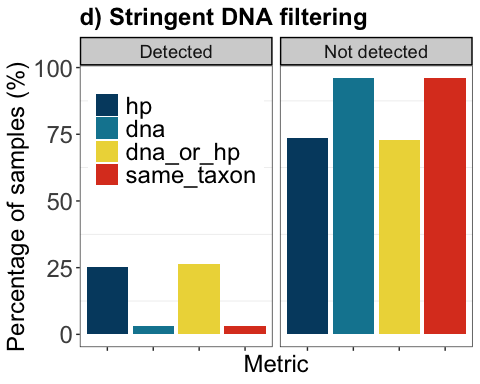
ggsave('figure2b\_birddetecthigh.jpeg', plot = bird\_fig2b, width = 7.5, height = 5, dpi = 300)

#Penguin detections medium stringency DNA QC  
  
#Colour palette #select the first four colours in fig2.pal to match the colours in previous graphs for "hp", "dna", "dna\_or\_hp" and "same\_taxon"  
#check palette using  
#str(fig2.pal)  
fig3.pal <- c("#00496F", "#0F85A0", "#EDD746", "#DD4124")  
  
#Figure 3a  
pingu\_fig3a <- ggplot(penguin\_detect\_med, aes(x=metric, y=percentage, fill=metric)) +  
 geom\_bar(stat="identity",position="dodge")+  
 theme\_bw() + theme(panel.grid.major = element\_blank()) +  
 theme(plot.title=element\_text(face="bold", size=18))+  
 theme(axis.title=element\_text(size=18)) + theme(axis.text=element\_text(size=18)) +  
 theme(axis.text.x=element\_blank())+  
 theme(legend.text=element\_text(size=18)) + #theme(legend.title=element\_text(size=14)) +  
 theme(legend.position=c(0.2,0.75), legend.justification=c(0.4,0.5))+  
 scale\_fill\_manual(values=fig3.pal)+  
 guides(fill=guide\_legend(title=NULL))+  
 labs(title ="c) Standard DNA filtering", x="Metric", y="Percentage of samples (%)") +  
 facet\_grid(.~detection)+  
 theme(strip.text = element\_text(size=rel(1.25)), #face="bold",   
 strip.background = element\_rect(fill="lightgrey", colour="black",  
 size=1))  
pingu\_fig3a



ggsave('figure3a\_penguindetectmed.jpeg', plot = pingu\_fig3a, width = 7.5, height = 5, dpi = 300)

#Penguin detections medium stringency DNA QC  
  
#Figure 3b  
pingu\_fig3b <- ggplot(penguin\_detect\_high, aes(x=metric, y=percentage, fill=metric)) +  
 geom\_bar(stat="identity",position="dodge")+  
 #ggtitle("a) Standard DNA filtering")+  
 theme\_bw() + theme(panel.grid.major = element\_blank()) +  
 theme(plot.title=element\_text(face="bold", size=18))+  
 theme(axis.title=element\_text(size=18)) + theme(axis.text=element\_text(size=18)) +  
 theme(axis.text.x=element\_blank())+  
 theme(legend.text=element\_text(size=18)) + #theme(legend.title=element\_text(size=14)) +  
 theme(legend.position=c(0.2,0.75), legend.justification=c(0.4,0.5))+  
 scale\_fill\_manual(values=fig3.pal)+  
 guides(fill=guide\_legend(title=NULL))+  
 labs(title ="d) Stringent DNA filtering", x="Metric", y="Percentage of samples (%)") +  
 facet\_grid(.~detection)+  
 theme(strip.text = element\_text(size=rel(1.25)), #face="bold",   
 strip.background = element\_rect(fill="lightgrey", colour="black",  
 size=1))  
pingu\_fig3b



ggsave('figure3b\_penguindetecthigh.jpeg', plot = pingu\_fig3b, width = 7.5, height = 5, dpi = 300)

#bird\_detect\_anova <- aov(value ~ variable, data = birdD\_long)  
#anova(bird\_detect\_anova)

## Including Plots

You can also embed plots, for example:



Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.