Phenome imputation of a population of outbred rats

Hannah Meyer

2018-07-19

Contents

The following vignette shows the imputation workflow for an additional dataset. More detailed information about the analysis steps can be found in the vignette 'Phenome imputation of a panel of prototrophic haploid yeast segregants'.

1. Dataset

The dataset consists of an outbred population of rats descended from eight inbred progenitors. 2,006 outbred rats were phenotypes for 195 traits of biomedical relevance including hematological, immunological, cardiovascular and fear-related phenotypes. 1,407 of the rats were genotyped [1],[2].

The phenotype and genotype data was downloaded from arrayexpress and figshare. The pre- processing of the data is described in detail in the vignette Pre-processing rat data The kinship was estimated from these genotypes: the [*NrSamples* x *NrSNPs*] genotypes X were filtered for minor allele frequency of at least 5% and the kinship estimated as $R = \frac{1}{NrSNPs} X^T X$.

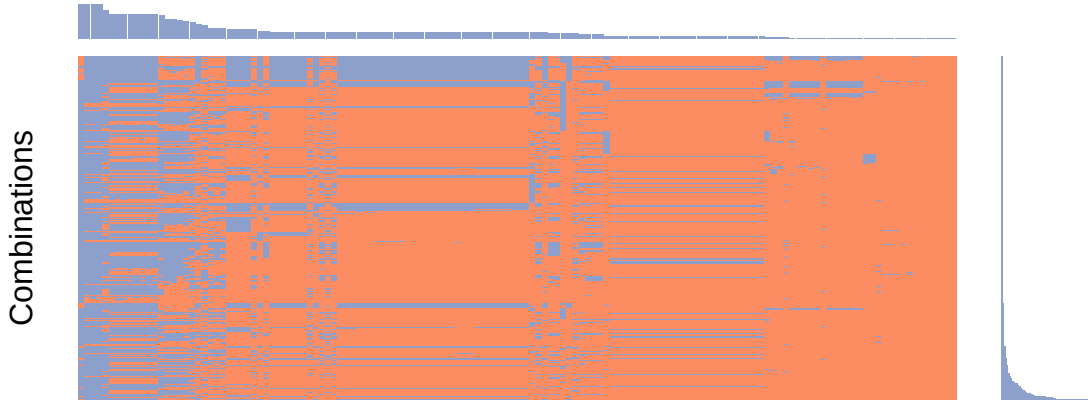
```
## libraries ####
library('ggplot2')
## directories ####
rawdir <- "~/data/LiMMBo/rat/rawdata/arrayexpress"
directory <- "~/data/LiMMBo/rat/processeddata"

## data ####
phenotypes_normal <- read.table(paste(directory, "/phenotypes_normal.csv",
                                      sep=""),
                                header=TRUE, row.names=1, stringsAsFactors=FALSE,
                                sep=",")

covariates <- read.table(paste(directory, "/covariates.csv", sep=""),
                                header=TRUE,  row.names=1, stringsAsFactors=FALSE,
                                sep=",")
```

```
kinship <- read.table(paste(rawdir, "/HS_rats_kinship_norm.csv", sep=""),
                            header=TRUE, stringsAsFactors=FALSE,
                            sep=",")

common_samples <- colnames(kinship)[colnames(kinship) %in%
                                      rownames(phenotypes_normal)]
kinship <- kinship[colnames(kinship) %in% common_samples,
                   colnames(kinship) %in% common_samples]
phenotypes_normal <- phenotypes_normal[rownames(phenotypes_normal) %in%
                                         common_samples,]
phenotypes_normal <- phenotypes_normal[match(colnames(kinship),
                                         rownames(phenotypes_normal)),]

covariates <- covariates[rownames(covariates) %in%
                                         common_samples,]
covariates <- covariates[match(colnames(kinship),
                                         rownames(covariates)),]

# for imputation purposes, combine covariate and phenotype data
combined_all <- cbind(covariates, phenotypes_normal)

## general parameters ####
cutoff <- 0.95
col <- c('#fc8d62','#8da0cb')
text_size <- 12
```

2. Pattern of missing data

Aggregation plot (middle) where all existing combinations of missing (blue) and non-missing (orange) values in the traits are depicted. The bar chart on its right shows the frequencies of occurrence of the different combinations. The histogram on the top shows the frequency of missing values for each trait.

```
## a) distribution
plot_pattern_missingness(combined_all, directory=directory,
                         name="missing_data_pattern_allTraits")
```




```
frequency_missingness <- data.frame(missing=
                                      apply(combined_all, 2, function(x)
```

```
                                              length(which(is.na(x)))/length(x)))
frequency_missingness$complete <- 1 - frequency_missingness$missing

Traits2Keep <- frequency_missingness$missing <= 0.20
combined_filterTraits <- combined_all[, Traits2Keep]

per_sample_missingness <- data.frame(missing=
                                       apply(combined_filterTraits, 1, function(x)
                                         length(which(is.na(x)))/length(x)))
per_sample_missingness$complete <- 1 - per_sample_missingness$missing

Samples2Keep <- per_sample_missingness$missing <= 0.20
kinship <- kinship[Samples2Keep, Samples2Keep]
combined <- combined_filterTraits[Samples2Keep,]
```

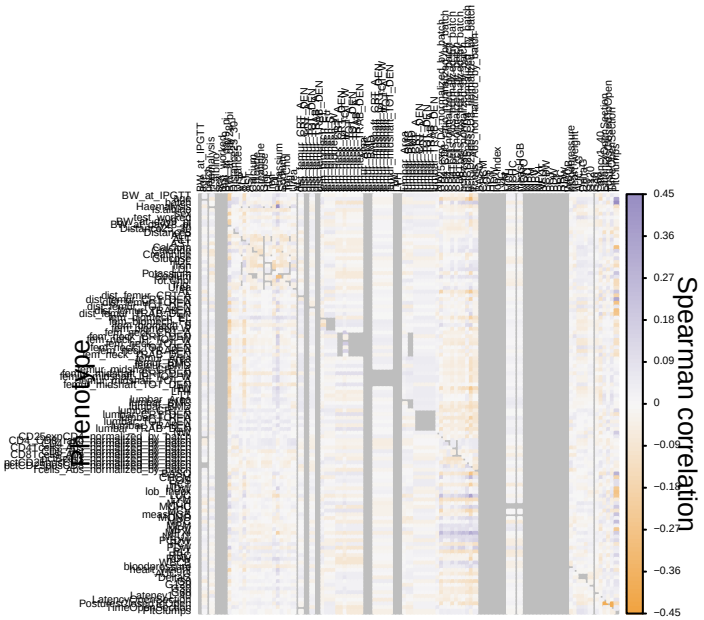
```
plot_pattern_missingness(combined, directory=directory,
                         name="missing_data_pattern_filteredTraits")
```

Any sample with more than 20% missing phenotype data is removed from further analyses, reducing the dataset to 1096 samples. The aggregation and frequency plots for this reduced dataset is shown above.

3. Missing data mechanism

Requirement for MAR data: missingness is predictable. If it can be demonstrated that one or more variables in the dataset are significantly correlated with missing values, missingness may be predictable Visiually examining predictable missingness by correlating the observed values across all samples with each column of an indicator matrix, i.e. the missingness patterns per trait. If all values were observed for a given trait, all values in the indicator matrix in this column were equal to zero and the correlation between the trait and the missingness was set to NA. Overall, there is sufficient evidence for predictable missingness.

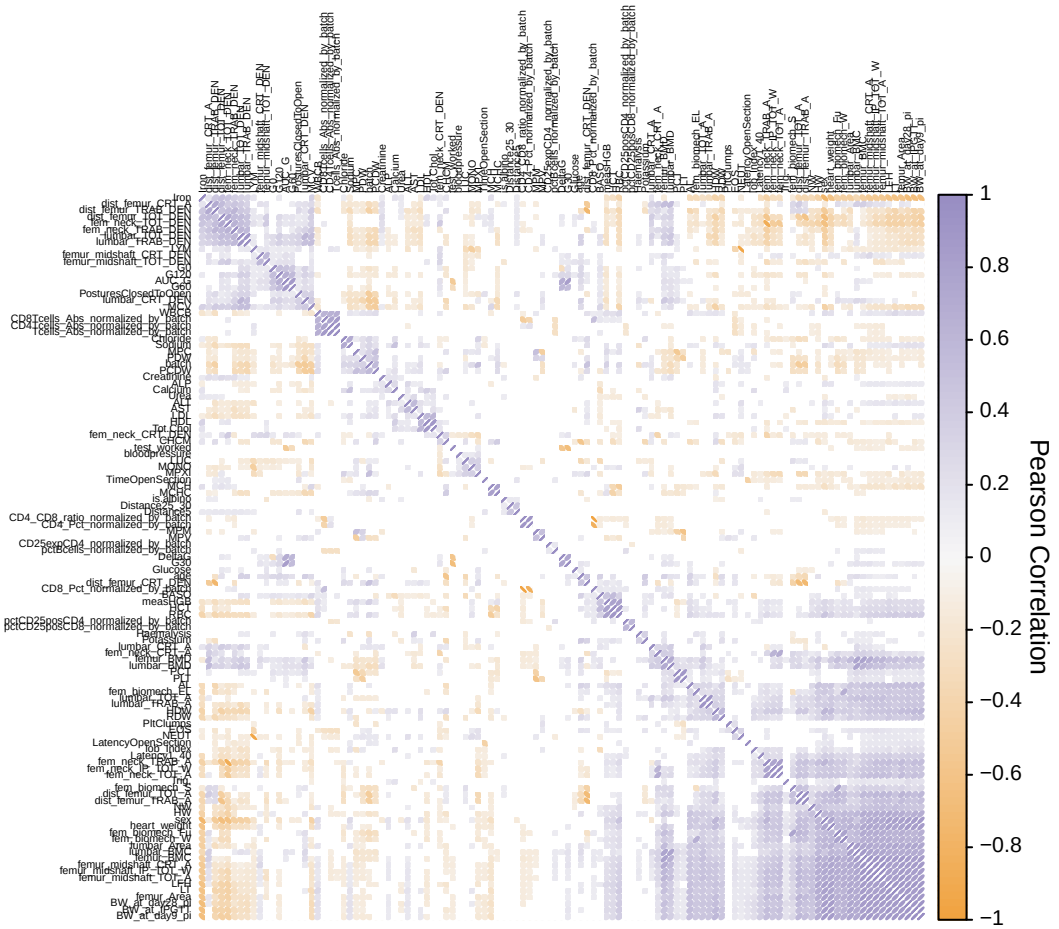
```
### MAR
corrMiss <- correlationMissingness(combined)
plot_correlation_missingness(data=corrMiss, savePdf=TRUE,
                             directory=directory, labelsize=0.3)
```



Missingness

4. Dataset with no missing values

```
combined_noNA <- fullPhenotypes(combined)

## a) correlation between phenotypes
corrPhenotypes <- correlationPhenotypes(combined)

plot_correlation_phenotypes(data_r=corrPhenotypes$r,
                            data_p=corrPhenotypes$padjust, savePdf=TRUE,
                            directory=directory, labelsize=0.3)
```



Out of the 1096 rats for which at least 80% of phenotypes were measured, 361 were fully phenotyped. The pairwise phenotype correlation across these rats is shown above.

5. Generate dataset with artificial missingness

Use the subset of the 361 fully phenotyped rats and introducing missing values with a similar pattern of missingness as observed in the original dataset. The results for the real and simulated dataset are similar in terms of frequencies and combinations of missing/non-missing traits.

```
combined_artificial <- artificialMissingness(data=combined, fulldata=combined_noNA,
                                             kinship=kinship, seed=3422)
plot_pattern_missingness(combined_artificial$data_addNA, directory=directory,
                         name="missing_data_pattern_simulated", savePdf=TRUE)
```

6. Impute artifically created missing data

The masked values in the dataset with artificial missingness were imputed with two genetic (phenix and mpmm) and two non-genetic (mvn and mice) methods. For imputation with mice, different predictor sets were analysed, based on the correlation of the measured phenotypes (All phenotypes as predictors, phenotypes with correlation > 0.1, 0.2 or greater 0.3). After imputation, the goodness of the imputation is evaluated by computing the Pearson correlation of the imputed values to the experimentally observed ones. Table 1 show those correlations across all traits for all imputation methods.

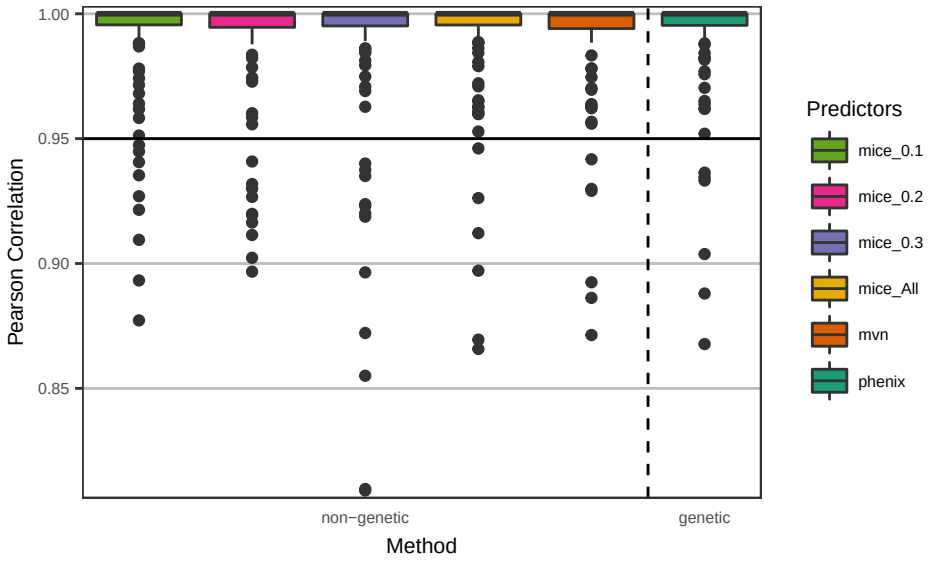
```
imputed <- imputeData(data=as.matrix(combined_artificial$data_addNA),
                      fulldata=combined_noNA,
                      kinship=combined_artificial$kinship,
                      method=c("phenix",  "mvn", "mice"),
                      cutoff=cutoff, testing=TRUE)
#> Warning in sqrt(1 - h * h): NaNs produced

#> Warning in sqrt(1 - h * h): NaNs produced
#> Using pheno as id variables
knitr::kable(imputed$summary,
caption="\\label{tab:imputation}Correlation of imputed vs real phenotype values")
```

Table 1: Correlation of imputed vs real phenotype values

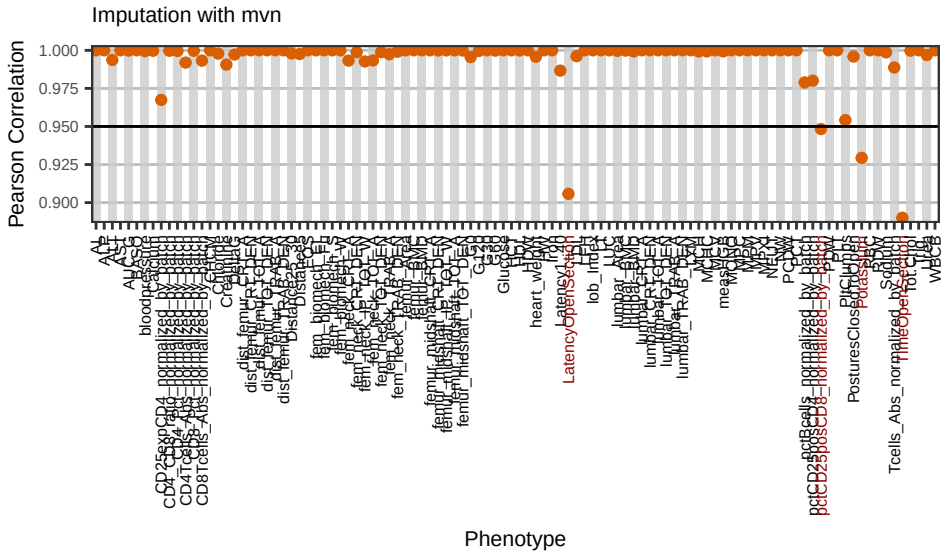
	median	mean	sd
mice_0.1	1	0.9899949	0.0231010
mice_0.2	1	0.9895366	0.0237503
mice_0.3	1	0.9860840	0.0355147
mice_All	1	0.9899801	0.0243124
mvn	1	0.9905609	0.0228132
phenix	1	0.9910931	0.0221556

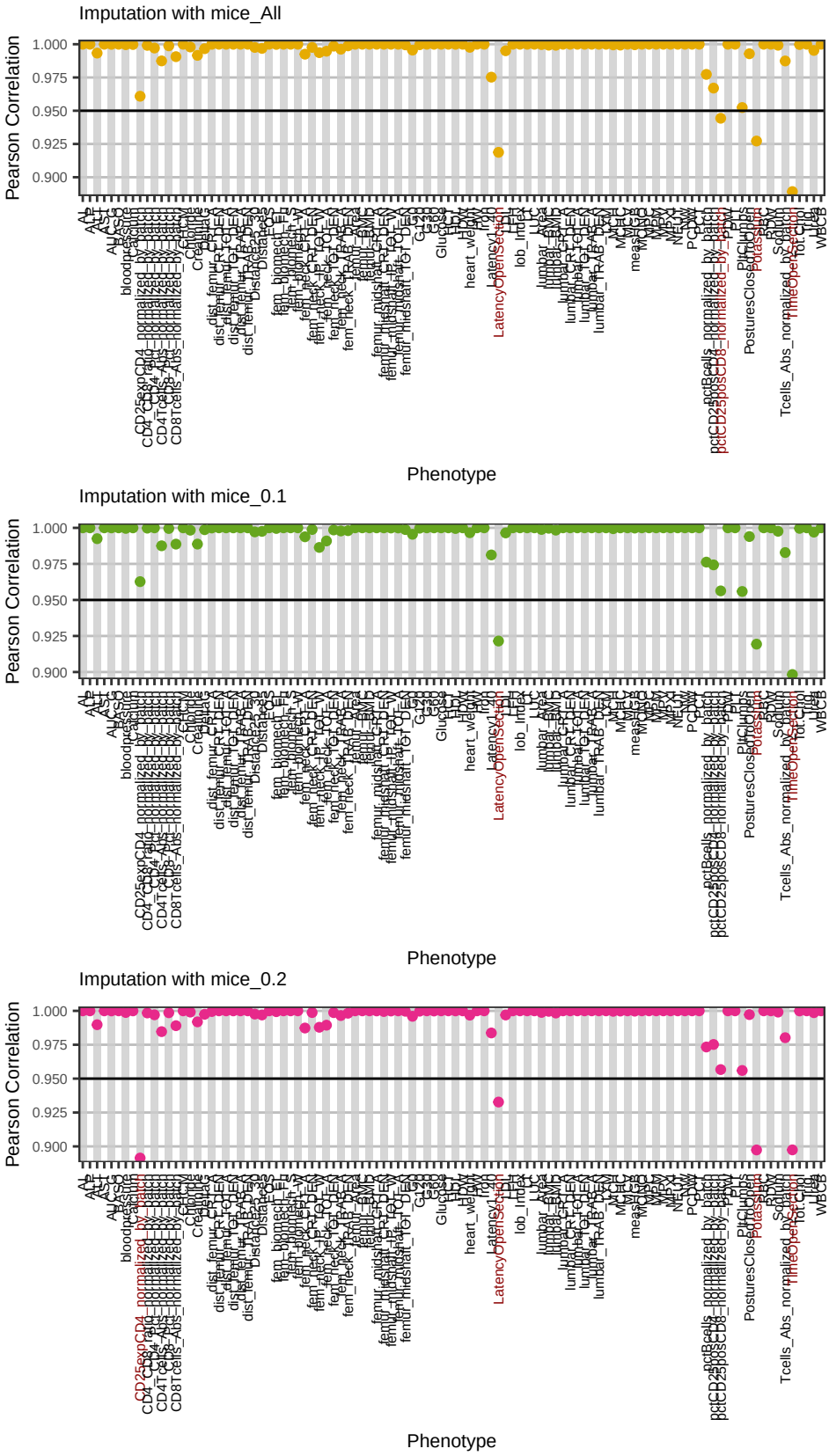
```
plot_overview_correlation_imputation(imputed$cor, savePdf=TRUE,
                                     directory=directory,
                                     text_size=6)
```

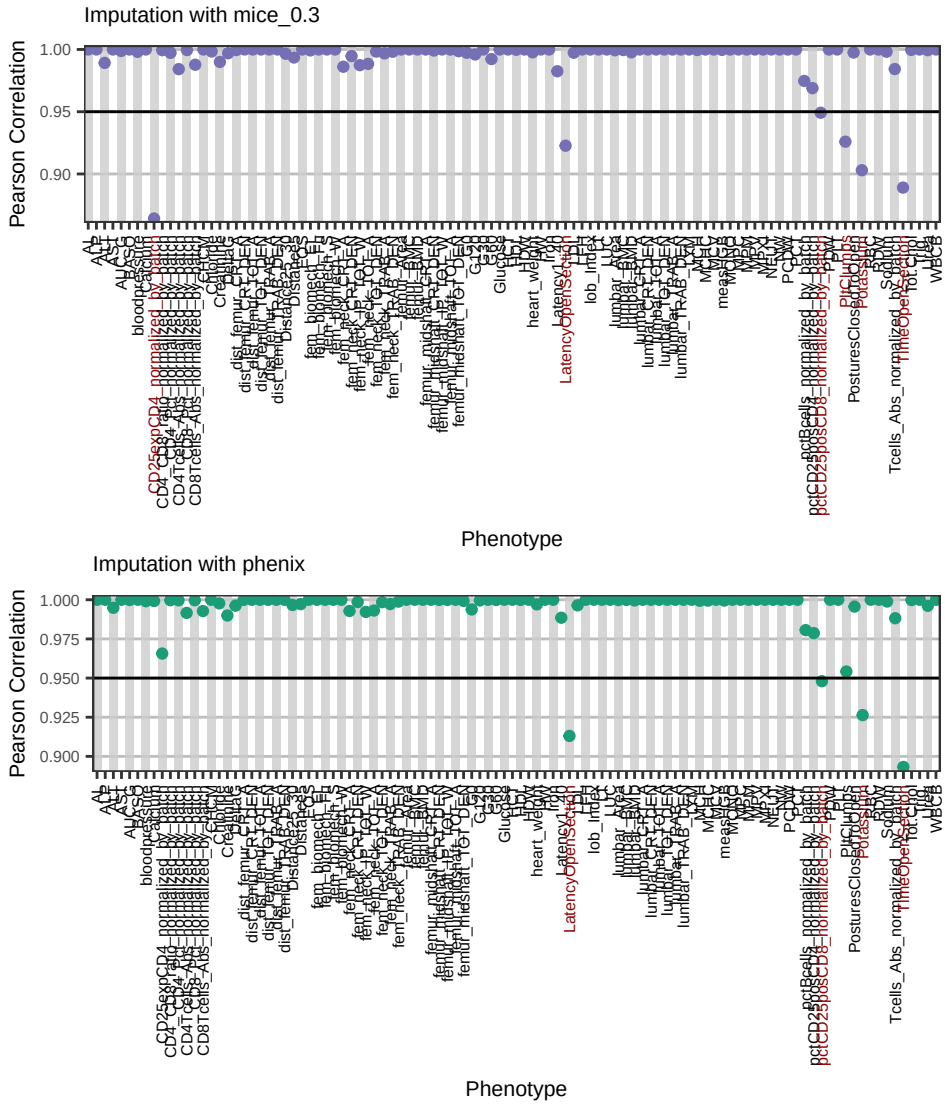
```
corr_plots_methods <- plot_individual_correlation_imputation(imputed$cor,
                                                             savePdf=TRUE,
                                       directory=directory, text_size=6)
```




```
dummy <- lapply(corr_plots_methods, function(p) print(p))
```




Imputation with mice_All
Pearson Correlation
1.000
0.975
0.950
0.925
0.900
Phenotype
Imputation with mice_0.1
Pearson Correlation
1.000
0.975
0.950
0.925
0.900
Phenotype
Imputation with mice_0.2
Pearson Correlation
1.000
0.975
0.950
0.925
0.900
Phenotype



7. Impute full data set

The traits that can be reliable imputed (correlation greater than 95%) are selected and the missing values from the phenotype set with 1096 rats imputed.

```
imputable <-  data.frame(sapply(c("phenix", "mvn", "mice"), imputableTraits,
                                imputed, cutoff))

imputed_phenos <- imputeData(data=combined, methods= c("phenix", "mice", "mvn"),
                        imputable=imputable, cutoff=0.95, testing=FALSE,
                        kinship=as.matrix(kinship))
saveRDS(imputed_phenos, paste(directory, "/imputed_phenotypes.rds", sep=""))
```

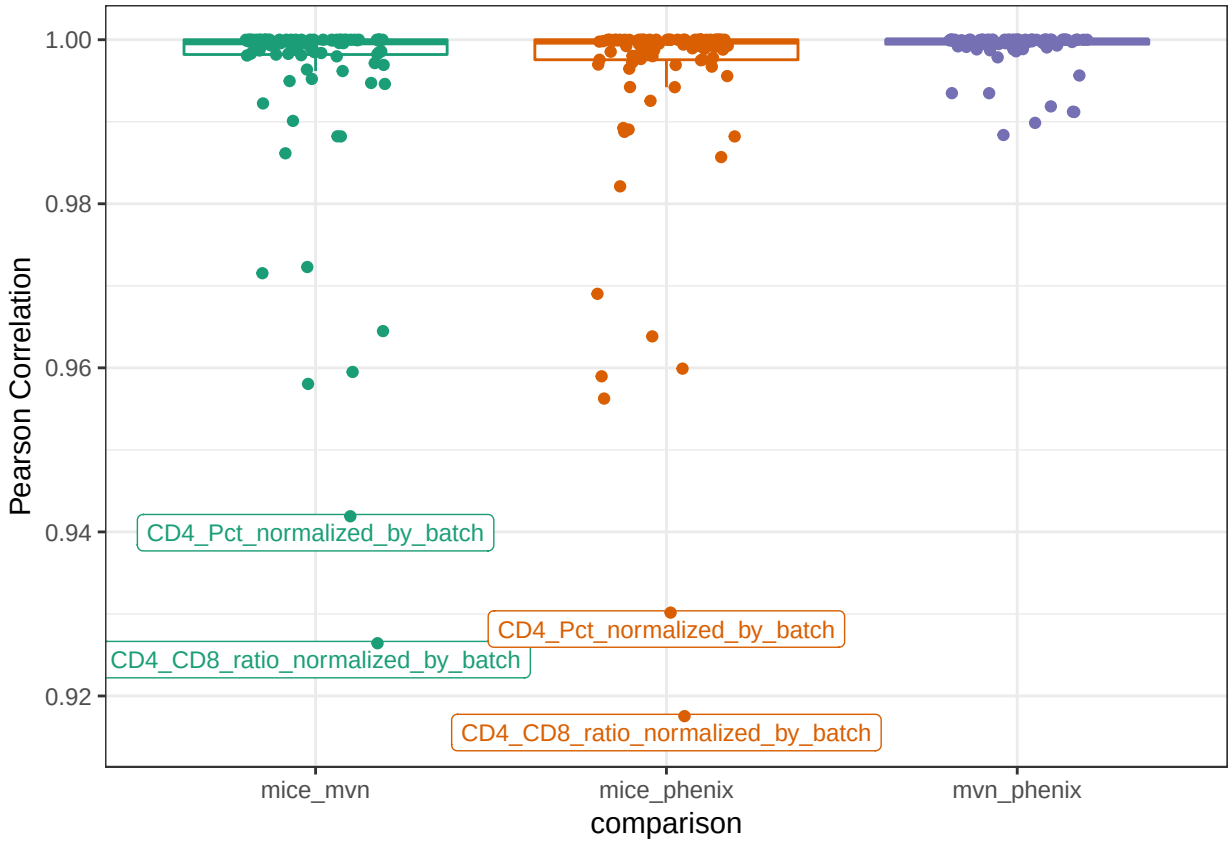
```
phenotype2covs <- readRDS(paste(directory, "/phenotype2covs.rds", sep=""))

regress <- function(trait, pheno, cov, pheno2cov ) {
    which.cov <- which(colnames(cov) %in% pheno2cov[[trait]])
```

```
    summary(lm(pheno[,colnames(pheno) == trait] ~ cov[,which.cov]))$residuals
}

combined_phenix <- imputed_phenos$imp$phenix$imp
pheno_phenix <- combined_phenix[, which(colnames(combined_phenix) %in% colnames(phenotypes_normal))]
cov_phenix <- combined_phenix[, which(colnames(combined_phenix) %in% colnames(covariates))]
phenix_reg <- sapply(colnames(pheno_phenix), regress, pheno=pheno_phenix,
                     cov=cov_phenix, pheno2cov=phenotype2covs)

write.table(phenix_reg, paste(directory, "/phenotypes_phenix_reg.csv", sep=""),
            sep=",", col.names=NA, row.names=TRUE, quote=FALSE)

combined_mvn <- imputed_phenos$imp$mvn$imp
pheno_mvn <- combined_mvn[, which(colnames(combined_mvn) %in% colnames(phenotypes_normal))]
cov_mvn <- combined_mvn[, which(colnames(combined_mvn) %in% colnames(covariates))]
mvn_reg <- sapply(colnames(pheno_mvn), regress, pheno=pheno_mvn,
                     cov=cov_mvn, pheno2cov=phenotype2covs)

write.table(mvn_reg, paste(directory, "/phenotypes_mvn_reg.csv", sep=""),
            sep=",", col.names=NA, row.names=TRUE, quote=FALSE)


combined_mice<- imputed_phenos$imp$mice$imp
pheno_mice <- combined_mice[, which(colnames(combined_mice) %in% colnames(phenotypes_normal))]
cov_mice <- combined_mice[, which(colnames(combined_mice) %in% colnames(covariates))]
mice_reg <- sapply(colnames(pheno_mice), regress, pheno=pheno_mice,
                     cov=cov_mice, pheno2cov=phenotype2covs)

write.table(mice_reg, paste(directory, "/phenotypes_mice_reg.csv", sep=""),
            sep=",", col.names=NA, row.names=TRUE, quote=FALSE)
```

The pairwise correlation of the phenotypes imputed with mice, mvn and phenix is shown below. In general, imputed phenotypes correlate strongly, while for two traits the correlation of the mice imputed traits versus imputation with mvn and phenix falls below a correlation of 0.95.

```
commonImputable <- intersect(intersect(colnames(mice_reg), colnames(mvn_reg)),
                             colnames(phenix_reg))

mice_mvn <- do.call(rbind, lapply(commonImputable, function(x) {
    tmp <- Hmisc::rcorr(mice_reg[,colnames(mice_reg) == x],
                        mvn_reg[,colnames(mvn_reg) == x])
    return(data.frame(p=tmp$P[1,2], r2=tmp$r[1,2], comparison="mice_mvn", trait=x))
}))

mice_phenix <- do.call(rbind, lapply(commonImputable, function(x) {
    tmp <- Hmisc::rcorr(mice_reg[,colnames(mice_reg) == x],
                        phenix_reg[,colnames(phenix_reg) == x])
    return(data.frame(p=tmp$P[1,2], r2=tmp$r[1,2], comparison="mice_phenix", trait=x))
}))
#> Warning in sqrt(1 - h * h): NaNs produced

phenix_mvn <- do.call(rbind, lapply(commonImputable, function(x) {
    tmp <- Hmisc::rcorr(phenix_reg[,colnames(phenix_reg) == x],
                        mvn_reg[,colnames(mvn_reg) == x])
```

```
    return(data.frame(p=tmp$P[1,2], r2=tmp$r[1,2],comparison="mvn_phenix", trait=x))
}))
#> Warning in sqrt(1 - h * h): NaNs produced

compare_corr <- rbind(mice_mvn, mice_phenix, phenix_mvn)
compare_corr$comparison <- as.factor(compare_corr$comparison)

p <- ggplot(data=compare_corr, aes(x=comparison, y=r2,
                                   color=comparison))
p + geom_boxplot(outlier.colour = NA) +
    geom_label(data=dplyr::filter(compare_corr, r2 < 0.95), aes(y=r2,
                                                                x=comparison,
                                                                label=trait),
               nudge_y = -0.002, size=3) +
    geom_jitter(width = 0.2) +
    scale_color_manual(values=c('#1b9e77','#d95f02','#7570b3'), guide=FALSE) +
    ylab("Pearson Correlation") +
    theme_bw()
```



8. References

1. Baud A, Hermsen R, Guryev V, Stridh P, Graham D, McBride MW, et al. Combined sequence-based and genetic mapping analysis of complex traits in outbred rats. Nature Genetics. Nature Publishing Group; 2013;45: 767–775. doi:10.1038/ng.2644

2. Baud A, Guryev V, Hummel O, Johannesson M, Hermsen R, Stridh P, et al. Genomes and phenomes of a population of outbred rats and its progenitors. Scientific Data. Nature Publishing Group; 2014;1: 140011. doi:10.1038/sdata.2014.11