`dcSimulation.ipynb`

Simulation notebooke to show why the phenotypic variance by X chromosome is differed depending on dosage compensation (twice in male on FDC, twice in female on NDC).

In [ ]:
```
# import modules
import numpy as np
```

In [ ]:
```
# ad-hoc function
def std_mtx(mtx):
    return (mtx - np.mean(mtx, axis=0)) / np.std(mtx, axis=0)
```

1. One-SNP model

In [ ]:
```
# arguments
maf = 0.3
n_sample = 3000
var_x = 0.1
n_iter = 1000
```

In [ ]:
```
# haplotype
hap_x_male = np.random.binomial(1, maf, size=n_sample)
hap_x1_female = np.random.binomial(1, maf, size=n_sample)
hap_x2_female = np.random.binomial(1, maf, size=n_sample)
```

In [ ]:
```
# effect size of the SNP
effect_size = np.random.normal(loc=0.0, scale=np.sqrt(var_x))
print("Effect size of the SNP : {:.3f}".format(effect_size))
```

```
Effect size of the SNP : 0.175
```

In [ ]:
```
# Expectation
exp_x_male = maf * (1 - maf) * effect_size**2
exp_x_female_noInactivation = 2 * maf * (1 - maf) * effect_size**2
exp_x_female_inactivation = 0.5 * maf * (1 - maf) * effect_size**2
print("Expected variance by male = {:.5f}".format(exp_x_male))
print("Expected variance by female(no inactivation) = {:.5f}".format(exp_x_female_noInactivation))
print("Expected variance by female(random inactivation) = {:.5f}".format(exp_x_female_inactivation))
```

```
Expected variance by male = 0.00641
Expected variance by female(no inactivation) = 0.01282
Expected variance by female(random inactivation) = 0.00320
```

In [ ]:
```
# phenotypic variance by the SNP

# phenotype ~ male
geno_x_male = hap_x_male
geno_x_female = hap_x1_female + hap_x2_female

# phenotype ~ female, no inactivation
pheno_x_male = hap_x_male * effect_size
pheno_x_female_noInactivation = geno_x_female * effect_size

# phenotype ~ female, random inactivation
n_cells = 1000
pheno_x_female_inactivation = np.zeros_like(pheno_x_female_noInactivation)

for cell in range(n_cells):
    active_idx = np.random.binomial(1, 0.5)

    if active_idx == 0:
        pheno_x_female_inactivation += hap_x1_female * effect_size
    elif active_idx == 1:
        pheno_x_female_inactivation += hap_x2_female * effect_size

pheno_x_female_inactivation = pheno_x_female_inactivation / n_cells
```

In [ ]:
```
print("Phenotypic variance by male = {:.5f}".format(np.var(pheno_x_male)))
print("Phenotypic variance by female(no inactivation) = {:.5f}".format(np.var(pheno_x_female_noInactivation)))
print("Phenotypic variance by female(random inactivation) = {:.5f}".format(np.var(pheno_x_female_inactivation)))
```

```
Phenotypic variance by male = 0.00632
Phenotypic variance by female(no inactivation) = 0.01323
Phenotypic variance by female(random inactivation) = 0.00331
```

2. Genotype matrix

In [ ]:
```
n_sample = 2000
n_snp = {"autosome":2000,
         "chrX":2000}
var_dict = {"autosome":0.5,
            "chrX":0.1}
maf_dict = {"autosome":np.random.uniform(low=0.05, high=0.95, size=n_snp["autosome"]),
            "chrX":np.random.uniform(low=0.05, high=0.95, size=n_snp["chrX"])}
n_cells = 1000
```

In [ ]:
```
# effect size
effect_dict = {}
for chr_type in ["autosome", "chrX"]:
    per_snp_var = var_dict[chr_type] / n_snp[chr_type]
    effect_dict[chr_type] = np.random.normal(loc=0.0, scale=np.sqrt(per_snp_var), size=n_snp[chr_type])
```

In [ ]:
```
# expectation
pAutosome = maf_dict["autosome"]
varAutosome = np.sum(2 * pAutosome * (1 - pAutosome) * var_dict["autosome"]/ n_snp["autosome"])
pChrX = maf_dict["chrX"]
varChrX = np.sum(pChrX * (1 - pChrX) * var_dict["chrX"]/ n_snp["chrX"])
print("Phenotypic variance by Autosome = {:.5f}".format(varAutosome))
print("Phenotypic variance by chrX = {:.5f}".format(varChrX))
```

```
Phenotypic variance by Autosome = 0.18303
Phenotypic variance by chrX = 0.01801
```

In [ ]:
```
# generate haplotype
haplotypeMale = {}
haplotypeFemale = {}

for chr_type in ["autosome", "chrX"]:
    hap1Male = np.random.binomial(n=1, p=maf_dict[chr_type], size=(n_sample, n_snp[chr_type]))
    if chr_type == "chrX":
        hap2Male = np.zeros((n_sample, n_snp[chr_type]))
    else:
        hap2Male = np.random.binomial(n=1, p=maf_dict[chr_type], size=(n_sample, n_snp[chr_type]))
    haplotypeMale[chr_type] = [hap1Male, hap2Male]

    hap1Female = np.random.binomial(n=1, p=maf_dict[chr_type], size=(n_sample, n_snp[chr_type]))
    hap2Female = np.random.binomial(n=1, p=maf_dict[chr_type], size=(n_sample, n_snp[chr_type]))
    haplotypeFemale[chr_type] = [hap1Female, hap2Female]
```

In [ ]:
```
# phenotype ~ male
phenoMale = {}
for chr_type in ["autosome", "chrX"]:
    genoMale = np.sum(haplotypeMale[chr_type], axis=0)
    phenoMale[chr_type] = np.dot(genoMale, effect_dict[chr_type])

# phenotype ~ female, no inactivation
phenoFemaleNoInactivation = {}
for chr_type in ["autosome", "chrX"]:
    genoFemale = np.sum(haplotypeFemale[chr_type], axis=0)
    phenoFemaleNoInactivation[chr_type] = np.dot(genoFemale, effect_dict[chr_type])

# phenotype ~ female, inactivation
phenoFemaleInactivation = {}

for chr_type in ["autosome", "chrX"]:
    if chr_type == "autosome":
        genoFemale = np.sum(haplotypeFemale["autosome"], axis=0)
        phenoFemaleInactivation["autosome"] = np.dot(genoFemale, effect_dict["autosome"])
    elif chr_type == "chrX":
        phenoByX = np.zeros(n_sample)
        for si in range(n_sample):
            tmpHap1 = haplotypeFemale["chrX"][0][si, :]
            tmpHap2 = haplotypeFemale["chrX"][1][si, :]

            numCellactiveHap1 = np.sum(np.random.binomial(n=1, p=0.5, size=n_cells))
            numCellactiveHap2 = 1 - numCellactiveHap1

            phenoByactiveHap1 = np.dot(tmpHap1, effect_dict["chrX"])
            phenoByactiveHap2 = np.dot(tmpHap2, effect_dict["chrX"])

            tmpPheno = (numCellactiveHap1 * phenoByactiveHap1 + numCellactiveHap2 * phenoByactiveHap2) / n_cells

            phenoByX[si] = tmpPheno

        phenoFemaleInactivation["chrX"] = phenoByX
```

In [ ]:
```
for chr_type in ["autosome", "chrX"]:
    print("[{}]".format(chr_type))
    print("Phenotypic variance by male = {:.5f}".format(np.var(phenoMale[chr_type])))
    print("Phenotypic variance by female(no inactivation) = {:.5f}".format(np.var(phenoFemaleNoInactivation[chr_type])))
    print("Phenotypic variance by female(random inactivation) = {:.5f}".format(np.var(phenoFemaleInactivation[chr_type])))
```

```
[autosome]
Phenotypic variance by male = 0.17145
Phenotypic variance by female(no inactivation) = 0.17100
Phenotypic variance by female(random inactivation) = 0.17100
[chrX]
Phenotypic variance by male = 0.01934
Phenotypic variance by female(no inactivation) = 0.03947
Phenotypic variance by female(random inactivation) = 0.00950
```

In [ ]: