

Naam: Sjoerd Beetsma
Klas: V2A1
Studentnummer: 6189293

Opdracht

Maak een Monte Carlo Simulatie die meerdere keren (duizenden) de competitie speelt met gebruik van deze tabel van kansen. Hoe vaker je een competitie speelt, hoe nauwkeuriger je voorspelling wordt. Houd bij hoeveel punten elk team scoort in de competitie (3 punten voor winst, 1 punt voor gelijk, 0 voor verlies), en maak een overzicht (ranking) van de teams. (Gebruik je eigen geprogrammeerde Random Number Generator)

Maak een overzicht van de gemiddelden rank van de teams. D.w.z. bepaal je eindantwoord als een matrix, waarbij per team wordt aangegeven welke kans deze heeft om op die plek in het eindklassement te eindigen, op basis van de gesimuleerde spelletjes.

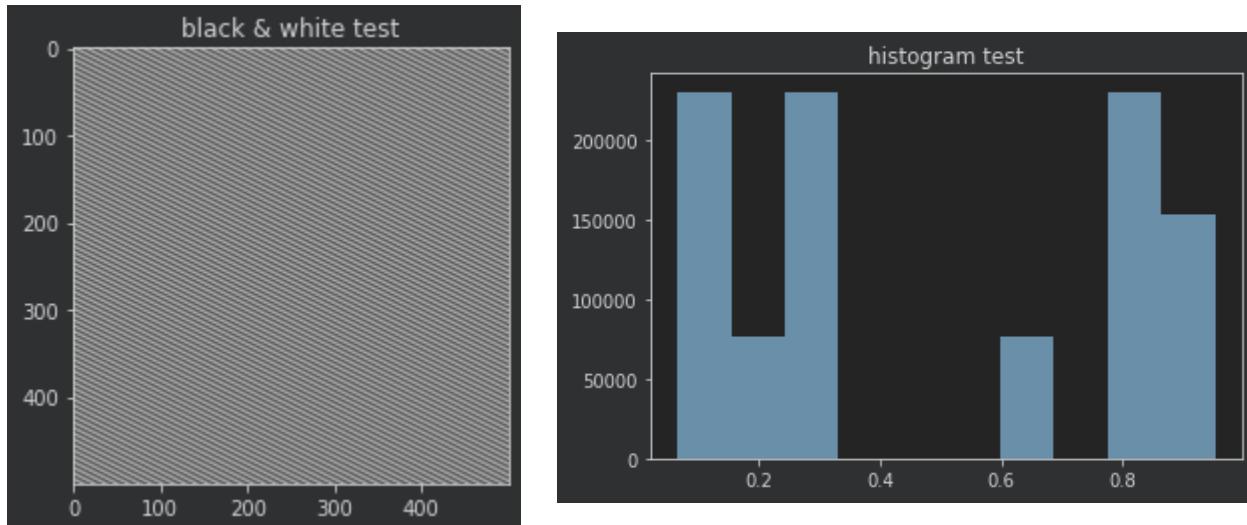
Verdiepend (voor een 8 of hoger): Om een betere, granuleerder beeld te schetsen van de waarschijnlijk uitkomst van een competitie, zal je ook doelpunten moeten bijhouden. Om dit voor elkaar te krijgen zal je de potjes die teams tegen elkaar spelen moeten uitbreiden: in plaats van alleen maar winst/gelijkspel/verlies te berekenen aan de hand van de kansen, moet je random gaan bepalen hoeveel doelpunten er gescoord zullen worden. Om dit te doen, kan je een Poisson-verdeling gebruiken.

Random number generators

Om de random number generators te testen zijn er histogrammen geplot met 1 miljoen samples en een zwart-wit plaatje gemaakt met 500*500 samples.

Middle square method

Middle square method geïmplementeerd zoals op Canvas.



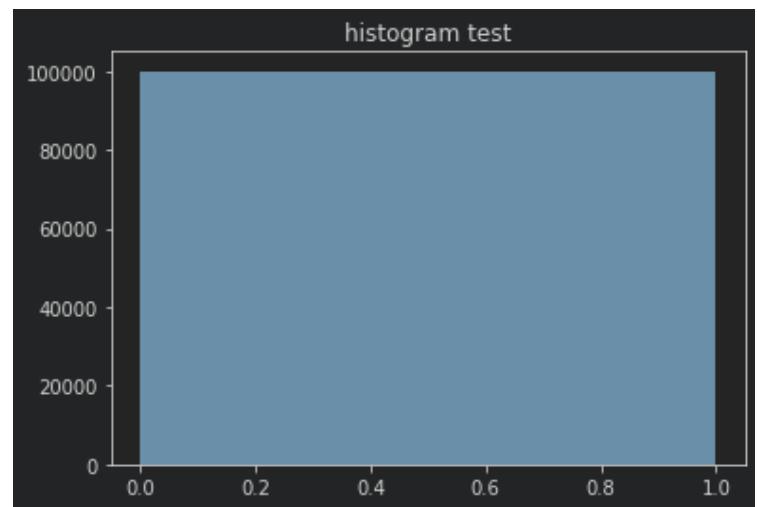
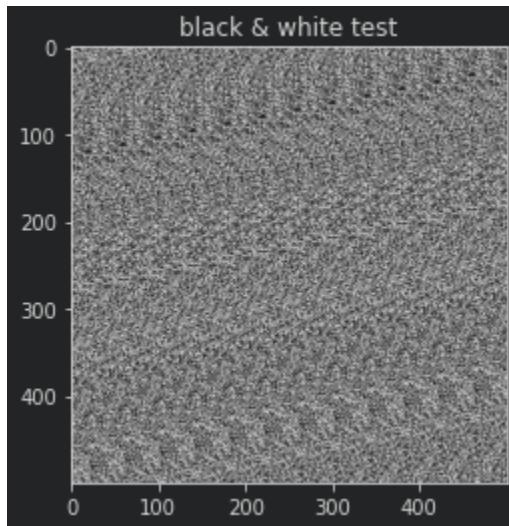
In het zwart-wit plaatje zijn niet veel ruis en er zijn duidelijke patronen.
De histogram is niet uniform.

Linear congenital generator (LCG).

Werkt met parameters: increment ‘c’, multiplier ‘a’ en modulus ‘m’.

Het algoritme past een lineaire formule toe op de seed: $a \times \text{seed} + c$ en volgt het met modulo m.

Dit wordt de nieuwe seed en de nieuwe seed gedeeld door m wordt het willekeurige getal tussen 0 en 1.



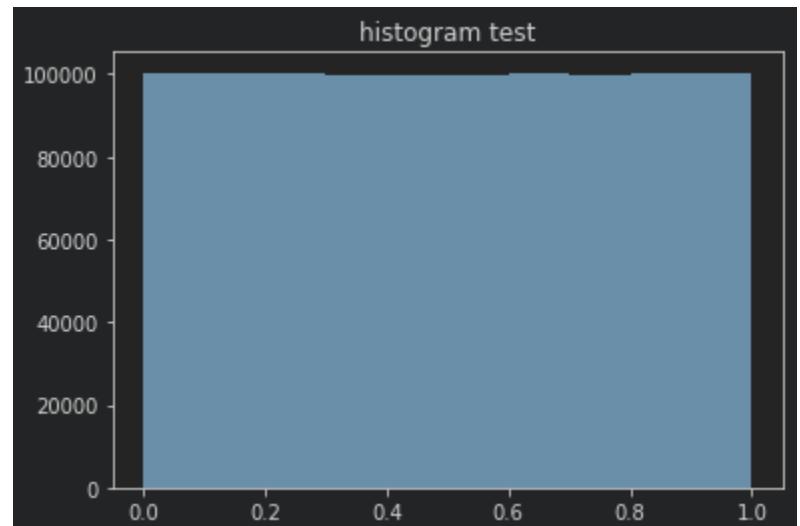
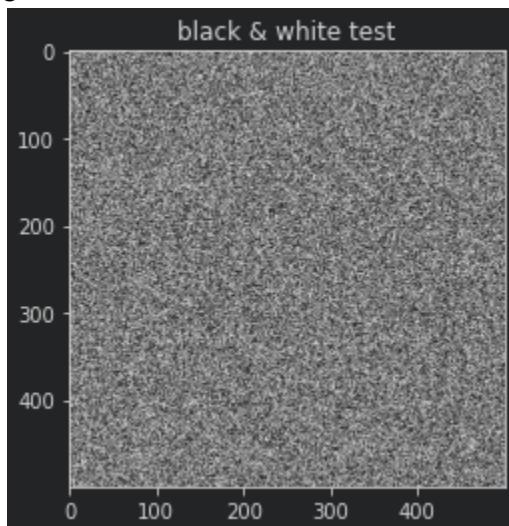
Deze generator presteert een stuk beter dan de Middle square method met een uniforme histogram en een zwart wit plaatje met veel ruis maar wel zichtbare patronen.

Lagged fibonacci generator (LFG)

Werkt met parameters: lag ‘j’, seed lengte ‘k’ en modulus ‘m’.

Het algoritme maakt met behulp van een gekozen seed en LCG eerst een seed aan van lengte k. Dit omdat LFG beter presteert met een “willekeurige” seed. En slaat deze op als een lijst.

Om een willekeurig nummer tussen 0 en 1 te genereren wordt er een binaire operatie gedaan met index j (de lag) in de seed en index k (het laatste getal) van de seed. In de implementatie is ‘+’ als binaire operatie gekozen.



Deze generator maakt ook een uniforme histogram, en maakt ook het zwart-wit plaatje met de meeste ruis tot zo ver, en zonder zichtbare patronen.

Conclusie RNG

Op basis van het testen van de verdeling via een histogram en de zichtbare patronen in een gegenereerd zwart-wit plaatje heb ik gekozen om LFG te gebruiken als Random number generator in de Monte Carlo simulatie.

Monte Carlo simulatie

Op basis van win, gelijkspel, en verlies kansen

De kansen-tabel is in een csv bestand gezet en wordt ingeladen als dataframe.

Per wedstrijd wordt er beslist aan de hand van de kansen wat het resultaat zal zijn als volgt;

Er wordt een willekeurig getal gegenereerd tussen 0 en 1.

Vervolgens wordt voor elk mogelijk resultaat en zijn kans de cumulatieve kans bijgehouden.

Wanneer de cumulatieve kans hoger is dan het willekeurige getal tussen 0 en 1 is, wordt het mogelijke resultaat definitief.

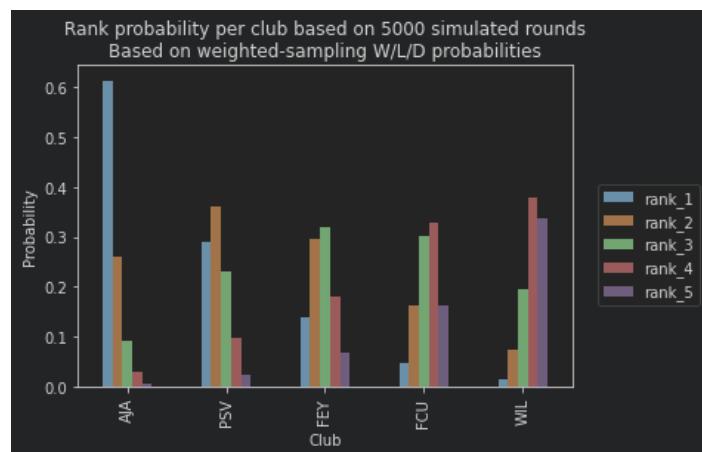
Volgens deze methode zijn 5000 competities gesimuleerd.

home	away			
		win	draw	lose
AJA	FEY	0.65	0.17	0.18
	PSV	0.54	0.21	0.25
	FCU	0.74	0.14	0.12
	WIL	0.78	0.13	0.09
	FEY	0.30	0.21	0.49
	PSV	0.37	0.24	0.39
FEY	FCU	0.51	0.22	0.27
	WIL	0.60	0.21	0.19
	PSV	0.39	0.22	0.39
	AJA	0.54	0.22	0.24
PSV	FCU	0.62	0.20	0.18
	WIL	0.62	0.22	0.16
	AJA	0.25	0.14	0.61
	FEY	0.37	0.23	0.40
FCU	PSV	0.29	0.24	0.47
	WIL	0.52	0.23	0.25
	AJA	0.17	0.18	0.65
	FEY	0.20	0.26	0.54
WIL	PSV	0.23	0.24	0.53
	FCU	0.37	0.25	0.38

Resultaten

Kansen per club op een bepaalde rank

	rank_1	rank_2	rank_3	rank_4	rank_5
AJA	61.34%	26.08%	9.04%	2.92%	0.62%
FEY	13.76%	29.66%	31.92%	17.92%	6.74%
PSV	29.1%	36.12%	22.96%	9.62%	2.2%
FCU	4.6%	16.12%	30.08%	32.82%	16.38%
WIL	1.56%	7.34%	19.58%	37.84%	33.68%



Op basis van verwachte goals en Poisson distributie.

De gemiddelde goals/tegen-goals als een team thuis of uit speelt worden ook via een csv bestand ingeladen als dataframe.

	HS	HC	AS	AC
AJA	3.2	0.9	3.1	0.6
FEY	2.4	1.1	2.2	0.8
PSV	2.1	0.7	1.8	1.3
FCU	1.9	1.2	3.0	2.4
WIL	1.4	1.7	1.0	1.5

Per wedstrijd wordt er besloten wat de verwachte goals voor de thuis en uit club zijn.

Voor de thuisclub is dit $(HS + AC) / 2$ en voor de uitclub $(HC + AS) / 2$. Bijv wanneer Ajax thuis speelt: $(3.2 + 1.3) / 2 = 2.25$.

Ook voor deze methode wordt gebruik gemaakt van cumulatieve kansen hiervoor wordt een willekeurige getal tussen 0 en 1 gegenereerd.

Vervolgens worden de kansen voor 0 tot een onbepaald aantal goals berekend met behulp van de Poisson verdeling. De Poisson verdeling berekent de kans op n goals als: verwacht aantal goals tot de macht n goals keer de constante e tot de macht -verwacht aantal goals gedeeld door n goals faculteit.

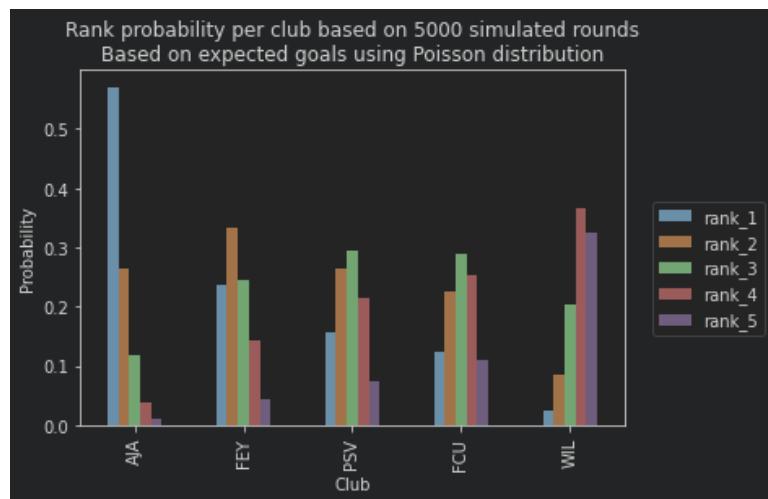
Bron: [Newcastle United: Using Poisson's Distribution to "Predict" Results of Upcoming Games \(A "Real Case"\) | by The Scientist | Medium](#). Waarbij voor elke kans de cumulatieve kans wordt bijgehouden. Wanneer de cumulatieve kans hoger is dan het willekeurige getal tussen 0 en 1 is, wordt het het aantal goals definitief. Dit wordt gedaan voor beide teams om zo het resultaat van de wedstrijd te bepalen a.d.h.v. het aantal gemaakte goals.

Ook volgens deze methode zijn 5000 competities gesimuleerd.

Resultaten

Kansen per club op een bepaalde rank

	rank_1	rank_2	rank_3	rank_4	rank_5
AJA	57.08%	26.34%	11.72%	3.9%	0.96%
FEY	23.68%	33.32%	24.58%	14.16%	4.26%
PSV	15.6%	26.28%	29.3%	21.54%	7.28%
FCU	12.28%	22.68%	28.82%	25.36%	10.86%
WIL	2.42%	8.36%	20.26%	36.56%	32.4%



Conclusie

De uitkomsten van de simulatie a.d.h.v. de win, gelijkspel en verlies kansen en de simulatie a.d.h.v. de verwachte aantal goals met de Poisson verdeling lijken erg op elkaar. Voor het grootste deel komen de kansen overeen binnen enkele procenten met enkele uitzonderingen.

Voor Ajax lijken alle kansen ongeveer overeen te komen met een maximale afwijking van een paar procent.

Voor Feyenoord zijn de kansen voor Rank 1 en Rank 3 in de verschillende simulaties aanzienlijk afwijkend.

Voor PSV lijken alle kansen aanzienlijk veranderd te zijn waarbij PSV beter presteert in de simulatie gebaseerd op W/D/L kansen.

Ook de kansen van FC Utrecht zijn redelijk anders met een betere prestatie in de simulatie gebaseerd op de Poisson verdeling.

Voor Willem II verschillen de kansen in beide simulaties het minst.

Het verschil in uitkomsten kan komen doordat de wedstrijd resultaten gebaseerd worden op verschillende kansen. In de eerste simulatie wordt het resultaat bepaald op W/D/L kansen van een bepaalde thuisclub tegen een specifieke uitclub. In de tweede simulatie wordt gebruik gemaakt van gemiddelde goals/tegen-goals van een bepaalde club wanneer deze uit of thuis speelt. Dit gemiddelde is niet gebaseerd op wedstrijden tussen 2 specifiek clubs zoals bij de W/D/L kansen.

Waarschijnlijk zal de simulatie gebaseerd op W/D/L kansen accurater zijn mits het aantal wedstrijden waar de kansen op gebaseerd zijn even veel is als het aantal wedstrijden waar de gemiddelde goals gebaseerd op zijn.

Dit is echter onrealistisch, het is waarschijnlijker dat er veel meer en recentere data beschikbaar is over het aantal gemiddelde goals waardoor de Poisson verdeling het in de werkelijkheid accurater zal doen in het voorspellen van een voetbal-competitie uitkomt.

	rank_1	rank_2	rank_3	rank_4	rank_5
AJA	61.34%	26.08%	9.04%	2.92%	0.62%
FEY	13.76%	29.66%	31.92%	17.92%	6.74%
PSV	29.1%	36.12%	22.96%	9.62%	2.2%
FCU	4.6%	16.12%	30.08%	32.82%	16.38%
WIL	1.56%	7.34%	19.58%	37.84%	33.68%

W/D/L Ranking kansen

	rank_1	rank_2	rank_3	rank_4	rank_5
AJA	57.08%	26.34%	11.72%	3.9%	0.96%
FEY	23.68%	33.32%	24.58%	14.16%	4.26%
PSV	15.6%	26.28%	29.3%	21.54%	7.28%
FCU	12.28%	22.68%	28.82%	25.36%	10.86%
WIL	2.42%	8.36%	20.26%	36.56%	32.4%

Poisson Ranking gebaseerd op doelsaldos