

Handover folosind ML în 5G

Project Sponsor: UPT, Lasting Software

Project Manager: Cornel Balint

Realizat de:

Preda Bogdan, An 3 TST RO

Dobosan Robert, An 3 TST EN

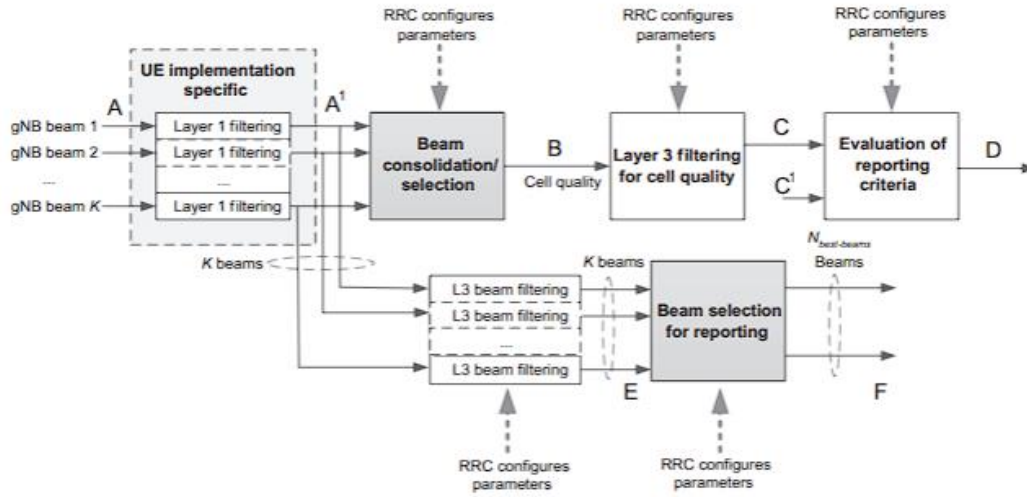


Figure 2.30
NR RRM measurement model [8].

The UE mobility state is determined, if the parameters (T_{CRmax} , N_{CR_H} , N_{CR_M} , and $T_{CRmaxHyst}$) are broadcast in SI for the serving cell. The **state detection criteria** are based on the following principles [12]:

- Normal-mobility state criterion: If the number of cell reselections during time period T_{CRmax} is less than N_{CR_M} .
- Medium-mobility state criterion: If the number of cell reselections during time period T_{CRmax} is greater than or equal to N_{CR_M} but less than or equal to N_{CR_H} .
- High-mobility state criterion: If the number of cell reselections during time period T_{CRmax} is greater than N_{CR_H} .

1.Motivație

Proiectul „5G Handover using ML” își propune să exploreze și să implementeze un model de handover în rețelele 5G utilizând tehnici de învățare automată. Prin acest proiect, echipa de dezvoltare va determina parametrii critici ai handover-ului, va selecta algoritmi AI potriviți și va crea un prototip care să optimizeze tranzițiile între celule pentru a îmbunătăți calitatea experienței utilizatorilor și eficiența rețelei.

Utilizarea tehnologiei Machine Learning în optimizarea handover-ului în rețelele 5G promite îmbunătățiri semnificative în gestionarea și predictibilitatea transferurilor de sesiune între celulele rețelei. Acest lucru este crucial pentru suportarea unui număr tot mai mare de dispozitive mobile și pentru aplicații ce necesită o lățime de bandă mare și o latență redusă.

Obiectivele Proiectului

- Identificarea Parametrilor de Handover: Definirea și analiza parametrilor cheie care influențează deciziile de handover în rețelele 5G.
- Selecția Tehnologiei de AI: Alegerea algoritmilor de învățare automată cei mai eficienți pentru predicția și optimizarea handover-ului.
- Prototipizarea și Testarea: Dezvoltarea unui prototip pentru a testa și valida modelul de handover asistat de AI.
- Evaluarea Performanței: Analiza performanței modelului de handover în scenarii reale de utilizare pentru a determina impactul asupra calității serviciului.

2. Studiu bibliographic

- 5G NR Architecture.pdf [Dahlman,_Erik_5G] (carte – pdf)
- Artificial Intelligence in 5G Technology (carte -pdf)
- Deep Reinforcement Learning based Handover Management for Millimeter Wave Communication (carte – pdf)
- HO ML (HandOver MachineLearning – carte)
- Google (cautari generale, Matlab, Inteligenta artificiala, Handover, Antene, Sateliti, 5G, 4G, ...)
- ChatGPT (Metode de rezolvare, Secvente cod, Explicatii cod Ai)

3. Proiectare

Pentru proiectarea acestui program de machine learning pentru predictive handover implică următoarele etape:

- **Identificarea tipurilor de handover care urmează să fie prezise**
- **Stabilirea indicatorilor de performanță (KPI) care sunt utilizați ulterior pentru evaluarea succesului modelului** (acuratețe, precizie, recall, confusion matrix, F1 score etc.).
- **Determinarea volumului și tipului de date necesare pentru antrenare, testare, utilizare ulterioară.**
- **Colectarea datelor** (precis în programul nostru sunt generate, pentru scopul antrenării, datele pentru încărcătura celulei, RSRP (putere), RSRQ (calitate) și decizia handover aleasă, după o anumită logică modelată cu ajutorul studiului bibliografic și discuției cu profesorul coordonator).

- **Preprocesarea datelor** pentru a le transforma în variabile numerice ce pot fi utilizate mai departe în program.
- **Explorarea și analiza corelațiilor între variabile** prin vizualizări grafice, plots și calcularea statisticilor descriptive pentru a rezuma datele și a identifica problemele.
- **Dezvoltarea modelului de Machine Learning**, alegând algoritmul potrivit: Random Forest.
- **Antrenarea modelului folosind datele pregătite**
- **Împărțirea setului de date în subseturi de antrenament și testare**
- **Utilizarea tehnicilor de validare încrucișată (cross validation)** pentru a evalua modelul.
- **Analiza și ajustarea modelului în funcție de rezultate**
- **Implementarea interfeței grafice intuitive**, folosind librării pentru aplicații web.
- **Integrarea componentelor de input pentru colectarea datelor necesare** (locație, putere, calitatea semnalului, cell load).
- **Implementarea vizualizării rezultatelor predicțiilor**, care oferă informații despre performanța modelului în timp real.
- **Integrarea finală a modelului de machine learning cu interfața grafică**
- **Testarea comunicării între componente**
- **Documentarea întregului proces.**
- **Prezentarea documentației și prezentare practică a programului.**

4. Testare

Pentru a asigura buna funcționare a modelului și îndeplinirea obiectivelor de performanță stabilite, testarea modelului include utilizarea diverselor metrice de evaluare pentru a determina acuratețea și eficiența predicțiilor:

Am împărțit datele în două seturi:

- **Set de antrenament:** folosit pentru a antrena modelul.
- **Set de testare:** folosit exclusiv pentru evaluarea finală a performanței modelului pe date necunoscute.

Am folosit și tehnica k-fold cross-validation, un proces ce implică împărțirea setului de date în subgrupe (fold-uri) și antrenarea modelului pe k-1 fold-uri și testarea pe fold-ul rămas. La final se calculează media performanțelor pentru toate fold-urile pentru a obține k-score (biblioteca scikit-learn implementează toate metricile folosite).

Metricile de evaluare, fiecare oferind o altă perspectivă asupra performanței, includ:

- **Acuratețe:** procentajul de predicții corecte din totalul predicțiilor. Este o metrică generală, dar poate fi înșelătoare în cazul dataset-urilor dezechilibrate.
- **Precizie:** proporția predicțiilor pozitive corecte din totalul predicțiilor pozitive realizate de model.
- **Recall** (Rata de recuperare): proporția predicțiilor pozitive corecte din totalul cazurilor pozitive existente în setul de date.
- **Confusion Matrix:** o matrice care arată numărul de predicții corecte și incorecte împărțite pe fiecare clasă, oferind o imagine detaliată a performanței modelului.

- **F1 Score:** media armonică a preciziei și recall-ului, utilă în cazul dataset-urilor dezechilibrate pentru a oferi o măsură echilibrată a performanței modelului.

Aceste metrice asigură o evaluare comprehensivă a modelului și ajută la identificarea punctelor forte și slabe ale acestuia, facilitând ajustările necesare pentru optimizarea performanței.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Unde TP, TN, FP, FN reprezintă adevaratele pozitive, adevaratele negative, falsurile pozitive și falsurile negative

Precizie:

Procentajul de predicții pozitive corecte din totalul predicțiilor pozitive, este util în cazul în care numărul falsurilor pozitive este ridicat.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Sensibilitate:

Procentajul de cazuri pozitive corect identificate din totalul cazurilor pozitive reale. Este importantă atunci când numărul falsurilor negative este ridicat.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-Score:

Media armonică a preciziei și a sensibilității, oferind un echilibru între acestea. Este utilă pentru a evalua performanța modelului în prezența unui echilibru între precizie și sensibilitate.

$$F1\text{-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

5. Implementare, Comentariu asupra implementarii logicii programului Machine Learning.

1. GENERARE DE DATE INITIALE

Table 2.4: Cell selection parameters [7,12].

Parameter	Description
<i>Srxlev</i>	Cell selection receive-level value (dB)
<i>Squal</i>	Cell selection quality value (dB)
<i>Qoffset_{temp}</i>	An offset temporarily applied to a cell (dB)
<i>Qrxlevmeas</i>	Measured cell receive-level value (RSRP)
<i>Qqualmeas</i>	Measured cell quality value (RSRQ)
<i>Qrxlevmin</i>	Minimum required receive-level in the cell (dBm). If the UE supports SUL frequency for this cell, <i>Qrxlevmin</i> is obtained from <i>RxLevMinSUL</i> , if present, in <i>SIB1</i> , <i>SIB2</i> , and <i>SIB4</i> . If <i>QrxlevminoffsetcellSUL</i> is present in <i>SIB3</i> and <i>SIB4</i> for the candidate cell, this cell-specific offset is added to the corresponding <i>Qrxlevmin</i> to achieve the required minimum receive-level in the candidate cell; otherwise, <i>Qrxlevmin</i> is obtained from <i>q-RxLevMin</i> in <i>SIB1</i> , <i>SIB2</i> , and <i>SIB4</i> . If <i>Qrxlevminoffsetcell</i> is present in <i>SIB3</i> and <i>SIB4</i> for the candidate cell, this cell-specific offset is added to the corresponding <i>Qrxlevmin</i> to achieve the required minimum receive-level in the candidate cell
<i>Qqualmin</i>	Minimum required quality level in the cell (dB). If <i>Qqualminoffsetcell</i> is signaled for the concerned cell, this cell-specific offset is added to achieve the required minimum quality level in the candidate cell
<i>Qrxlevminoffset</i>	An offset to the signaled <i>Qrxlevmin</i> taken into account in the <i>Srxlev</i> evaluation as a result of periodic search for a higher priority PLMN while camped normally in a VPLMN
<i>Qqualminoffset</i>	Offset to the signaled <i>Qqualmin</i> taken into account in the <i>Squal</i> evaluation as a result of periodic search for a higher priority PLMN while camped normally in a VPLMN
<i>P_{compensation}</i>	If the UE supports the additional <i>P_{max}</i> in the <i>NR-NS-PmaxList</i> , if present, in <i>SIB1</i> , <i>SIB2</i> , and <i>SIB4</i> , $\max(P_{EMAX1} - P_{PowerClass}, 0) - [\min(P_{EMAX2}, P_{PowerClass}) - \min(P_{EMAX1}, P_{PowerClass})]$ (dB) else $\max(P_{EMAX1} - P_{PowerClass}, 0)$ (dB)
<i>P_{EMAX1}</i> , <i>P_{EMAX2}</i>	Maximum transmit-power level that a UE may use when transmitting in the uplink in the cell (dBm) is defined as <i>P_{EMAX}</i> . If the UE supports SUL frequency for this cell, <i>P_{EMAX1}</i> and <i>P_{EMAX2}</i> are obtained from the <i>p-Max</i> for SUL in <i>SIB1</i> and <i>NR-NS-PmaxList</i> for SUL, respectively, in <i>SIB1</i> , <i>SIB2</i> , and <i>SIB4</i> ; otherwise, <i>P_{EMAX1}</i> and <i>P_{EMAX2}</i> are obtained from the <i>p-Max</i> and <i>NR-NS-PmaxList</i> , respectively, in <i>SIB1</i> , <i>SIB2</i> , and <i>SIB4</i> for regular uplink
<i>P_{PowerClass}</i>	Maximum RF output power of the UE (dBm) according to the UE power class

- Limitându-ne la Signal_Strength, Signal_Quality și Cell_Load.

RSRQ și RSRP sunt parametrii ai semnalului specifici celulei, specificați în 3GPP. RSRP este parametrul ce specifică puterea semnalului în dBm și este definit ca puterea medie (în wați) a elementelor ce poartă semnale de referință specifice celulei în banda considerată. RSRQ este parametrul ce specifică calitatea semnalului (în dB).

$$\text{RSRQ} = \frac{N \times \text{RSRP}}{\text{RSSI}}$$

unde RSSI e Carrier Received Signal Strength Indicator(indicator semnal primit de purtătoare)

- Cell_Load - random între 0 și 1 (100%).
- Signal_Strength: random, cu valori între -50 și -100 (dB)
 - dacă Signal_Strength este între -50 și -65, Signal_Quality va fi tot random între -5 și -10,
 - dacă Signal_Strength este între -65 și -85, Signal_Quality va fi tot random între -10 și -15,
 - dacă Signal_Strength este între -85 și -100, Signal_Quality va fi tot random între -15 și -20,
- pentru cazul 1 de mai sus: nu se face Handover (Handover_Decision = 0)
- pentru cazul 3 se face Handover (Handover_Decision = 1)
- pentru cazul 2 se pune Handover_Decision = 0 sau 1

pentru cazul 2 se consideră Handover_decision determinată de Cell_Load. Handover_decision = 0 dacă Cell_Load peste 0.5 și Handover_decision = 1 dacă Cell_Load este sub 0.5.

- Folosim pragurile $P1=75$ și $Q1=12.5$ (jumătate din RSRP și RSRQ) pentru scenariul A2 de handover (Serving cell becomes worse than a threshold).

Table 2.1 3GPP Specified Handover Events and Trigger Conditions [34], [35]

Event	Trigger Conditions
A1	Serving cell becomes better than a threshold
A2	Serving cell becomes worse than a threshold
A3	Neighbor cell becomes offset better than serving cell
A4	Neighbor cell becomes better than a threshold
A5	Serving cell becomes worse than threshold1 and neighbor cell becomes better than threshold2
A6	Neighbor cell becomes offset better than secondary cell (This event for carrier aggregation)
B1	Inter-RAT neighbor cell becomes better than threshold
B2	Serving cell becomes worse than threshold1 and Inter-RAT neighbor cell becomes better than threshold2

- Creez un tabel .csv cu cell load, RSRP, RSRQ și decizia handover. (1000 eșantioane)

2.ANTRENARE MODEL

- Citesc tabelul creat.
- Definesc coloanele de interes pentru decizia target (Handover) - Cell load, RSRP, RSRQ.
- Împart cele 1000 de eșantioane cu rația 0.2: 800 de antrenament și 200 de testare.
- Creez un obiect de tip RandomForestClassifier din biblioteca sklearn, în care poți modifica parametri precum câte decision trees să folosească (n_estimators) sau max_features, adică câte coloane random să ia în calcul de fiecare dată când

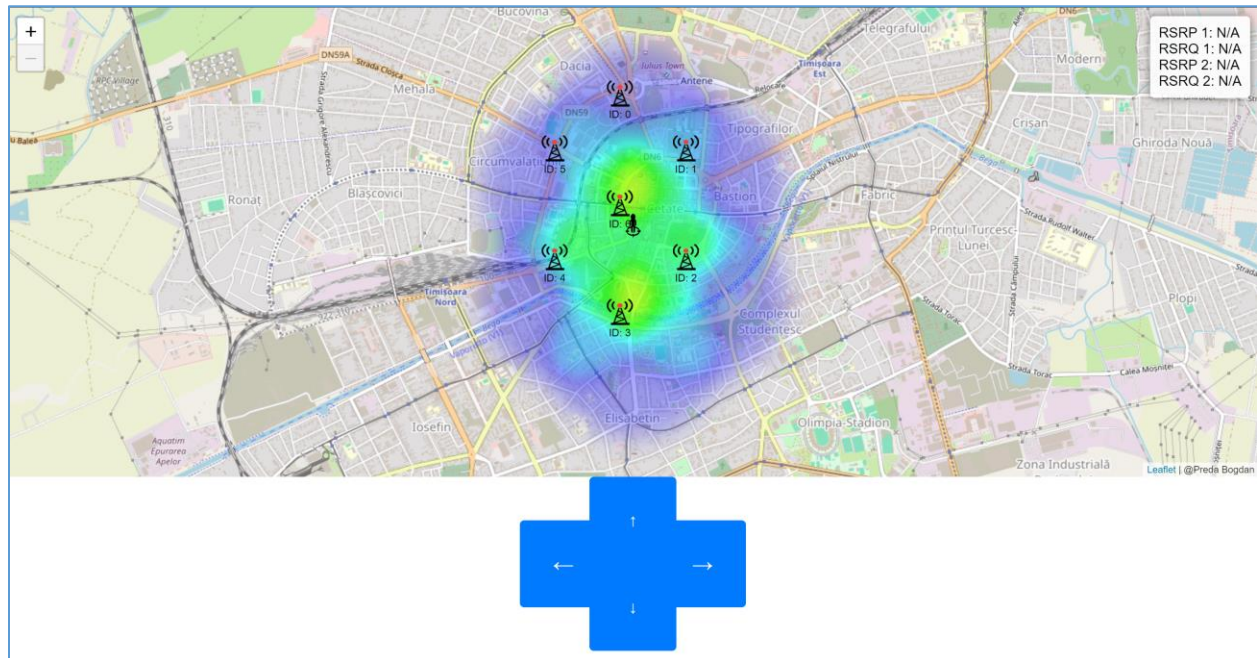
creează un decision tree(e pus ca radical din câte coloane sunt în total)

- Antrenăm modelul cu `model.fit(X_train,Y_train)` (y e `handover_decision` și x reprezintă restul coloanelor).
- Folosim diverse metode de a evalua modelul antrenat, comparând predicția acestuia cu `Y_test`.
- Salvăm modelul pentru a nu trebuii să îl antrenăm de fiecare dată.

3.UTILIZARE MODEL

- Deschidem modelul antrenat cu algoritmul Random Forest și datele noastre.
- Citim tabelul cu datele pe care dorim să le analizăm (în aceeași ordine coloanele ca și în tabelul cu date de antrenament, dar diferența cheie ca nu are coloana de `handover_decision`, știm doar `cell load`, `rsrp`, `rsrq` și nu știm dacă se dorește `handover` sau nu)
- Folosim modelul cu `loaded_model.predict(X)` (X reprezintă coloanele `cell load`, `rsrp`, `rsrq`) pentru oricât de multe eșantioane dorim, nu trebuie să fie multe pentru că modelul e deja antrenat, se poate și cu unul singur.
- Astfel, în variabila `predictions` avem o listă cu deciziile de `handover` calculate.
- Dacă avem și `handover_decision` real, îl putem compara cu cel calculat și observăm că modelul antrenat are o acuratețe foarte mare.

4.APLICATIE GRAFICA



Această aplicație simulează procesul de handover între antenele de telefonie mobilă într-un oraș specific. Ea utilizează hărți interactive pentru a vizualiza poziționarea și acoperirea antenelor, precum și pentru a urmări mișcarea unui utilizator virtual. Aplicația este construită pe baza librăriei Leaflet și oferă următoarele funcționalități:

Folosește hărțile OpenStreetMap pentru a oferi o vizualizare detaliată a zonei urbane.

Centrul hărții este stabilit la coordonatele (45.75372, 21.22571), corespunzătoare orașului Timișoara din România.

Antenele sunt plasate pe harta conform unui model circular, cu una dintre ele situată în centrul cercului și alte șase amplasate echidistant la periferia acestuia.

Fiecare antenă este marcată cu un icon personalizat și are un identificator unic (ID).

Un marker personalizat reprezintă utilizatorul și poate fi mutat pe hartă fie prin drag-and-drop, fie prin apăsarea butoanelor direcționale (nord, sud, est, vest) sau a tastelor corespunzătoare (W, A, S, D).

Poziția utilizatorului este folosită pentru a calcula intensitatea semnalului primit de la fiecare antenă.

Intensitatea semnalului este calculată pe baza distanței dintre utilizator și antene, utilizând un model simplificat care presupune că intensitatea scade liniar cu distanța.

Intensitatea semnalului este afișată pentru cele două antene cu cel mai puternic semnal.

Aplicația detectează și afișează mesajele de handover atunci când utilizatorul se deplasează din aria de acoperire a unei antene în aria de acoperire a alteia.

Dacă utilizatorul se află într-o zonă fără acoperire (intensitatea semnalului sub un anumit prag), se afișează un mesaj de pierdere a conexiunii.

O hartă de căldură vizualizează intensitatea semnalului în diferite zone ale orașului, ajutând la identificarea zonelor cu acoperire bună și slabă.

Aplicația este utilă pentru simularea și înțelegerea procesului de handover în rețelele de telefonie mobilă, oferind o interfață vizuală interactivă pentru explorarea acoperirii și calității semnalului în zone urbane.

5. Concluzie:

Proiectul „5G Handover using ML” demonstrează potențialul semnificativ al tehnologiilor de învățare automată în optimizarea proceselor de handover în rețelele 5G. Implementarea unui model de handover asistat de AI, utilizând parametri critici precum RSRP și RSRQ, a permis îmbunătățirea semnificativă a tranzițiilor între celule, asigurând o experiență superioară utilizatorilor și eficientizând gestionarea rețelei. Evaluările detaliate ale performanței modelului prin diverse metrice, inclusiv precizia, recall-ul și scorul F1, au evidențiat avantajele utilizării AI în acest context. Aplicația grafică dezvoltată a oferit o interfață intuitivă pentru vizualizarea și analizarea procesului de handover, facilitând astfel înțelegerea și aplicarea practică a modelului. Prin urmare, acest proiect reprezintă un pas important către adoptarea pe scară largă a tehnologiilor de machine learning în infrastructurile de comunicații mobile de nouă generație, subliniind rolul crucial al AI în îmbunătățirea performanței și fiabilității rețelelor 5G.

6. BIBLIOGRAFIE

1. E. Dahlman, S. Parkvall, J. Skold, and P. Beming, 5G NR: The Next Generation Wireless Access Technology. Academic Press, 2020.
2. Y. Li, M. Chen, W. Shi, and Z. Zhang, "Artificial Intelligence in 5G Technology," IEEE Access, vol. 6, pp. 50175-50185, 2018.
3. X. Zhang, J. Zheng, Y. Zhang, and M. Lu, "Deep Reinforcement Learning Based Handover Management for Millimeter Wave Communication," IEEE Transactions on Vehicular Technology, vol. 68, no. 2, pp. 1624-1635, Feb. 2019.
4. R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction. MIT Press, 2018.

5. 3GPP, "Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures," 3GPP TS 36.213 version 14.3.0 Release 14, 2017.
6. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.
7. T. Wang, L. Zhang, and Y. Zhang, "Machine Learning Applications in Cellular Networks: Connection Disruption Analysis and Handover Optimization," IEEE Wireless Communications, vol. 26, no. 1, pp. 104-110, Feb. 2019.
8. F. Y. Li, W. Zhang, and L. Hanzo, "Learning-aided Networking: A Study of Emerging Machine Learning Techniques in 5G Networks," IEEE Transactions on Communications, vol. 67, no. 2, pp. 1332-1351, Feb. 2019.
9. 3GPP (3rd Generation Partnership Project) - <https://www.3gpp.org/>
10. IEEE Xplore Digital Library - <https://ieeexplore.ieee.org/>
11. MathWorks (MATLAB & Simulink) - <https://www.mathworks.com/>
12. OpenStreetMap - <https://www.openstreetmap.org/>
13. Towards Data Science (Machine Learning and AI articles) <https://towardsdatascience.com/>
14. Scikit-learn (Machine Learning in Python) - <https://scikit-learn.org/>
15. Kaggle (Datasets and Machine Learning Competitions) - <https://www.kaggle.com/>
16. Leaflet (JavaScript library for interactive maps) - <https://leafletjs.com/>