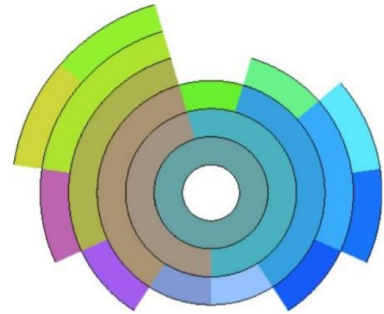


Visualization study materials - Q&A

Every question is worth 10 points, 25 points needed for an E (2018)

1. Radial tree - basic characteristics, converting a radial tree graph to a standard tree graph

Radial tree je metóda na zobrazovanie stromovej dátovej štruktúry. Tiež nazývaná "sunburst displays". Koreň hierarchie je umiestnený v strede radiálneho zobrazenia. Vrstvy sú reprezentované sústrednými prstencami. Každý prstenec je rozdelený (podľa počtu uzlov) v danej úrovni hierarchie. Podobná stratégia ako treemapy - počet koncových uzlov daného podstromu určuje veľkosť obrazovky, ktorá bude alokovaná pre tento podstrom. Na rozdiel od treemap zobrazujú radiálne techniky aj vnútorné uzly, nie najmä koncové.



Pseudokód:

Označení:

Start = počáteční úhel uzlu (iniciálně 0)

End = koncový úhel uzlu (iniciálně 360)

Origin = pozice středu radiálního zobrazení (např. [0, 0])

Level = aktuální stupeň hierarchie (iniciálně 0)

Width = tloušťka každého radiálního pásu - založena na maximální hloubce a velikosti displeje

sunburst(Node n, Start st, End en, Level l)

if n je koncový uzel (nemá potomky)

vykresli_radiální_sekci(Origin, st, en, l * Width, (l+1) * Width);

Return;

for each potomek n(child_i) získej počet koncových uzlů v podstromu;

sečti počet koncových uzlů;

spočti procentuální poměr koncových uzlů v každém podstromu(percent_i);

for each podstrom

spočti počáteční/koncový úhel na základě velikosti podstromů, jejich

uspořádání a rozsahu úhlů;

sunburst(child_i, st_i, en_i, l+1);

2. Dimension reduction techniques - PCA, MDS - choose one and describe in detail

Algoritmy na automatické znižovanie dimenzie datasetu, kde sa zachováva väčšina podstatných vlastností vstupnej množiny.

PCA - Principal Component Analysis

Vytvára doplnkové atribúty, ktoré sú lineárnou kombináciou pôvodných dátových premenných. Tieto atribúty definujú podpriestor premenných. Postup:

- Vyberieme priamku, ktorá pokrýva čo najviac vstupných dát - hlavná komponenta. Priamka prechádza ťažiskom a je ku všetkým dátam najbližšie ako sa dá.
- Vyberieme priamku, ktorá je kolmá na prvú priamku - tá tvorí druhú komponentu. Taktiež musí prechádzať ťažiskom.
- Opakujeme postup, kým nie sú spočítané všetky hlavné komponenty dimenzií, alebo kým nie je dosiahnutý požadovaný počet hlavných komponent

MDS (Multidimensional Scaling)

Snaží sa nájsť takú reprezentáciu vstupných dát v nižšej dimenzií, ktorá najlepšie zachová vzdialenosti medzi jednotlivými dátami pôvodnej vstupnej množiny. Postup:

- Spočítame vzdialenosti medzi každou dvojicou bodov v pôvodnom priestore. (pri n bodoch je to $n(n - 1)/2$ operácií)
- Prevedieme všetky body do priestoru požadovanej nižšej dimenzie (často náhodne prevedieme)
- Spočítame rozdiel medzi vzdialenosťami bodov v pôvodnom a novom priestore. Rozdiel sa nazýva **stress**
- Ak priemerný, alebo akumulovaný rozdiel medzi vzdialenosťami (stressom) bude menší ako userom definovaný prah, vrátime výsledok - done!
- Ak je väčší, ako userom definovaný prah - pre každý bod spočítame smerový vektor, označujúci smer kam by sa mal bod posunúť, aby sme redukovali rozdiel medzi vzdialenosťami daného bodu a všetkých ostatných bodov.
- Na základe týchto výpočtov transformujeme dátové body do cieľovej nižšej dimenzie – s ohľadom na spočítané vektory. Vrátime sa ku kroku 3 algoritmu.

3. Focus and context - describe a distortion that is used. When is it appropriate to use it?

Trieda interakcií spojená so štruktúrou grafu. Sú to techniky, kde je určitá podmnožina štruktúry prezentovaná detailne (**focus**), a zvyšok štruktúry je zobrazený len v obrysoch, aby mal user príslušný **kontext**.

Fish eye

Jedna z najpopulárnejších techník tohto typu je **rybie oko**, kde je “focus” časť zväčšená pomocou nelineárneho škálovania, zatiaľčo časť “kontext” je proporčne zmenšený. Takáto deformácia môže byť spravená v priestore obrazovky (čiže odvodená od pixelov), alebo v priestore štruktúry (odvodená od komponentu grafu). Druhý prípad je vhodný práve pre vizualizáciu grafov, kde môžeme napr. zväčšiť jednu vetvu stromu na úkor ostatných vetiev, alebo môžeme zvýrazniť všetky hrany vychádzajúce z daného uzlu, aby sme lepšie odhalili jeho susedné uzly.

Vhodné použitie: v interaktívnych vizualizáciách, kde chceme nechať užívateľa rozhodnúť, ktoré konkrétne detaily sú pre neho dôležité, bez toho, aby si musel príliš pamätať zvyšné údaje... Môže nastať problém s čitateľnosťou okolia, prípadne interpretácia okolia focusu môže byť nesprávna.

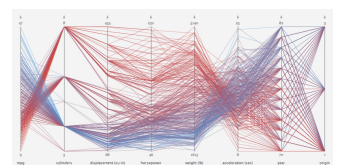


4. Rainbow scale - why is it a bad idea?

Vďaka nelineárnemu rozloženiu farieb môže zmeniť význam vizualizácie. Vytvára neexistujúce ohraničenia v obraze, ktoré nie sú prítomné v dátach, ale vznikajú kvôli významnému rozdielu medzi farbami susediacimi na tejto škále. Rainbow škála nie je uniformná s ohľadom na ľudské vnímanie.

5. Parallel coordinates

Využívajú sa na zobrazenie multivariate dát. Osy sú zobrazené paralelne za sebou a každá os reprezentuje jednotlivé usporiadanie danej dimenzie (dimenzia je daná počtom paralelných ôs). Dátový bod je polyčiara, ktorá pretína každú os podľa jej príslušiacich hodnôt.



6. How to deal with missing or “bad” values in input data

Je viacero možností, ako s takýmito chybami vo vstupných dátach bojovať. Riešenie takýchto problémov je zaradené v časti predspracovania dát.

- 1) Odstránenie chybného záznamu - jedna z najčastejších techník. Risk straty množstva záznamov.
- 2) Priradenie (pred)definovanej hodnoty - Každý premennej určíme hodnotu, ktorú priradíme v prípade, že bude záznam chybný alebo bude chýbať. Hodnotu určíme tak, aby vo vizualizácii bolo vidieť, ktoré data boli problematické.

- 3) Priradenie priemernej hodnoty - hodnotu nahradíme priemernou hodnotou danej premennej alebo danej dimenzie zo všetkých záznamov.
- 4) Priradenie hodnoty odvodennej od najbližšieho suseda - Vyberieme susedný záznam, ktorý je hodnotami v ostatných premenných najbližší záznamu s chýbajúcou alebo chybnou hodnotou. Následne, pomocou zmeny v ostatných premenných, nastavíme chýbajúcu či chybnú hodnotu.
- 5) Spočítanie náhradnej hodnoty - zložitý proces, treba dlhodobý výskum.)

7. Name visualisation evaluation methods (testing of usability, expert review, field test, case study). Describe at least 3

Expert review - odborník na vizualizácie/na dáta, ktoré má daná vizualizácia zobrazovať, zhodnotí, či vizualizácia spĺňa svoj účel

Field test - vizualizácia sa ukáže cieľovej skupine, a zisťuje sa, do akej miery bola vizualizácia správne interpretovaná a aký vplyv mala na userov. Testuje sa v kontexte reálneho použitia

Case study - plné použitie vizualizácie a jej detailná analýza - jej efektivita a expresivita, cena, zložitosť, správnosť predspracovania dát, interpretovateľnosť, naplnenie účelu.

Testing of usability - cieľová skupina skúša použiť danú vizualizáciu, analyzuje sa zrozumiteľnosť, intuitívnosť, správna interpretácia

8. What is chart junk?

Všetko, čo nie je nevyhnutnou súčasťou vizualizácie, čo nemá informačný význam a rozptyľuje užívateľa od správnej interpretácie.

Data loss spôsobená chart junkom

Nepotrebné vizuálne informácie môžu spôsobiť occlusion a potlačiť skutočnú (správnu) vizualizáciu. Musíme nájsť správny balans medzi estetickosťou a praktickosťou.

Rule of thumb: vizualizácia by mala poskytovať užívateľovi možnosť odstrániť on-demand vizuálne detaily, ktoré nie sú nevyhnutné.



9. Cartogram and its types

Je to zovšeobecnenie bežných tematických máp. Hodnoty tematickej mapovanej premennej nahrádzajú skutočnú plochu (land area) alebo vzdialenosť. Tým je geometria alebo priestor mapy skreslený, a namiesto poskytovania informácií o skutočnej veľkosti či tvare nejakého geografického celku, poskytuje dodatočné informácie namapovanej premennej.

Nespojitý kartogram

Vyhovuje obmedzeniam týkajúcich sa plochy a tvaru, nazachováva topológiu vstupnej mapy (vnútorné oblasti). Nedochádza k problémom vnímania mapy ako takej. Avšak pôvodná veľkosť a tvar mapy obmedzuje veľkosti vnútorných polygónov, a teda nie je možné ľubovoľne zväčšiť malé polygóny bez škálovania (zväčšenia) celej mapy. Dôležité oblasti sú tak ťažko viditeľné a priestor obrazovky nie je plne využitý.

Nepríľahlé/nesusediace kartogramy

Škálujú všetky polygóny na ich cieľové veľkosti, ktoré presne vyhovujú priestorovým požiadavkám. Tvary môžu byť jemne relaxované, polygóny sa tak dotýkajú ale bez prekryvu. Polygóny si neudržiavajú vzájomné “susedské” vzťahy. Dochádza ku strate globálneho tvaru a topológie mapy, čo zhoršuje vnímanie mapy ako celku.

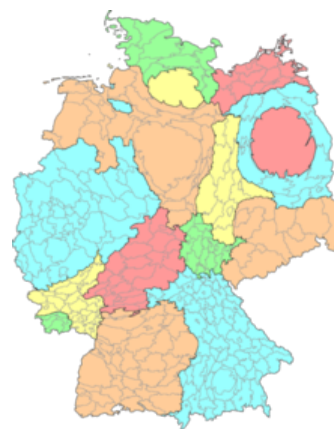
Kruhové kartogramy

Ignorujú tvar vstupných polygónov (vnútorných oblastí) a reprezentujú ich kruhmi. Väčšinou sú plošné aj topologické obmedzenia relaxované, vykazuje tak tento typ kartogramu rovnaké problémy, ako predošlí.

Spojité kartogramy

Spojité kartogramy pripomínajú pôvodnú mapu omnoho viac, než iné varianty kartogramov. Úplne zachovávajú topológiu mapy a relaxujú dané plošné a tvarové obmedzenia. Nedokážu plne zachovať tvar a plochu. Hľadanie kompromisu medzi zachovaním tvaru a plochy je teda pomerne zložitý optimalizačný problém, vytvoriť “bezchybný” kartogram (zachovávajúci tvar, orientáciu, susednosť), nie je obecne riešiteľný problém.

Tento problém môžeme definovať ako problém deformácie mapy. Vstupom je mapa (rovinná polygoniálna sieť) P , a sada hodnôt X , jedna pre každý región P . Cieľom je deformovať mapu tak, aby plocha každého regiónu odpovedala jeho hodnote X , a aby si regióny zachovali celkový tvar a boli rozpoznateľné.



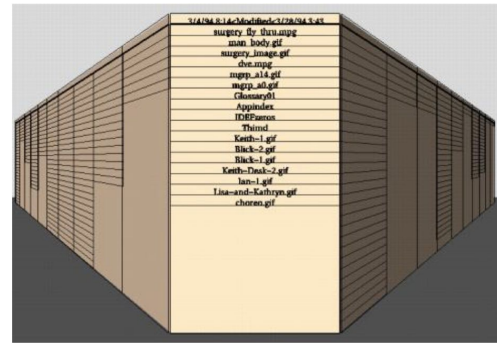
Kvalita automatizovaných algoritmov riešiacich tento problém je rôzna. Radiálne metódy často neposkytujú akceptovateľné výsledky, pretože tvary polygónov sú často silno deformované.

Obdĺžnikové kartogramy

Aproximácia známych máp pomocou obdĺžnikov. Nájdenie rozdelenia dostupného priestoru obrazovky, kde plochy obdĺžnikov proporcionálne odpovedajú štatistickým hodnotám. Obdĺžniky sú umiestňované čo najbližšie ich pôvodným polohám a susedom.

12. Perspective walls

Metóda pre lepšiu navigáciu vo vizualizácii obsahujúcej veľký počet dokumentov a dát. Zobrazuje sa jeden panel priamo viditeľný pozorovateľovi a ostatné panely sú otočené tak, akoby postupne mizli v diaľke. Steny sa používajú takým štýlom, že užívateľ s nimi môže interagovať a to tak, že sekvenčne prechádza "stránky" (panely steny) dopredu a dozadu.



13. Document visualisation - describe at least 3 types

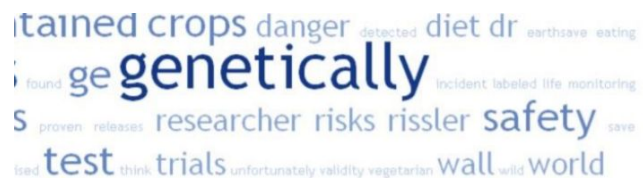
Korpus - sada dokumentov. Objekty (často považované za atomické) vnútri korpusov - slová, vety, odstavce, celé dokumenty, či ďalšie korpusy (aj obrázky a videá). Dokumenty sú často štruktúrované a obsahujú mnoho atribútov a metadát (daný formát, informácie o autorovi, dátumu vytvorenia, ...).

Práca s dokumentami:

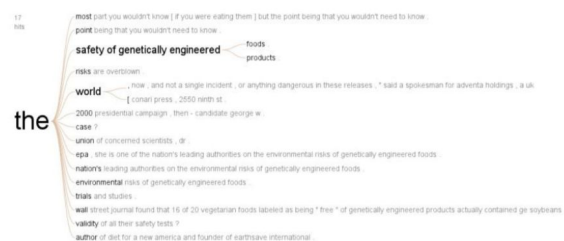
- Dotazovanie, kde výsledkom je relevantný dokument či jeho časť (vyhľadanie kľúčového slova)
- Počítanie štatistík o dokumentoch (počet slov, ich rozloženie, frekvencia)
- Identifikácia vzťahov medzi odstavcami či dokumentami v korpuse (hľadanie dokumentov týkajúcich sa chrípky)

Vizualizácie dokumentov

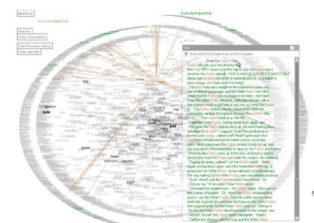
Tag clouds - Text clouds/word clouds - rozloženie jednotlivých tokenov, ktoré majú veľkosť a sú zafarbené na základe ich frekvencie v dokumente. Variáciou je napr. Wordle.



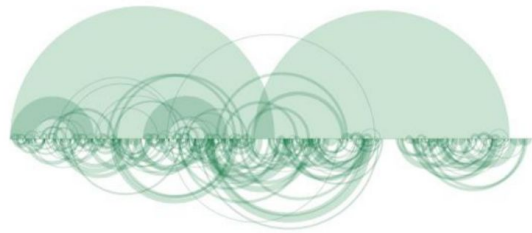
Word tree - vizuálna reprezentácia nie len frekvencia termov, ale aj ich kontextu. Veľkosť slúži k reprezentácii frekvencie termu alebo frázy. Koreňom stromu je užívateľom definované slovo (fráza) a vetvy reprezentujú rôzne kontexty, v ktorých je slovo (fráza) použité v dokumente.



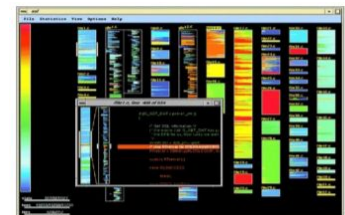
TextArc - reprezentácia rozloženia slov a ich konektivity. TextArc reprezentuje ako sa termy vzťahujú k riadkom, v ktorých sa objavujú. Každé slovo textu je vykreslené okolo elipsy. Slová s vyššou frekvenciou sú vykreslené väčšie a tmavšie/svetlejšie, a sú vykreslené vnútri elipsy v závislosti na ich výskyte na kružnici.



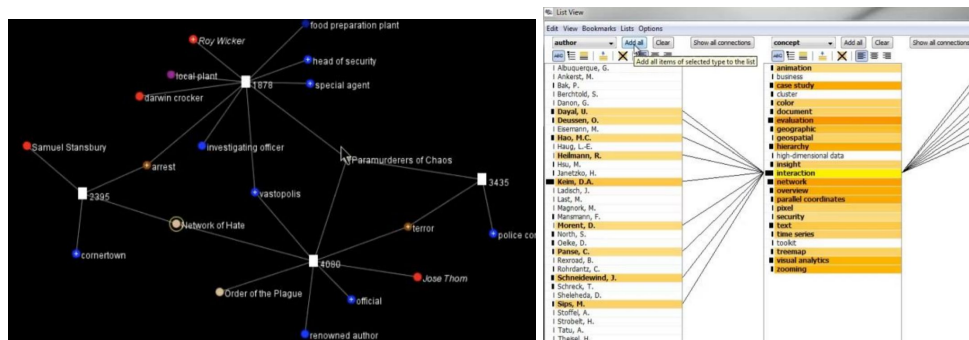
Arc Diagram - zameriava sa na opakujúci text či sekvenciu. Opakujúce sa podsekvencie sú spojené polkruhovými oblúkmi. Šírka oblúku reprezentuje dĺžku podsekvencie, výška vzdialenosť medzi podsekvenciami.



SeeSoft - software vizualizáciu zmien v dokumentoch (najmä v kóde) - "vek" riadku, nedávnu zmenu, počet zmien, ... Každý stĺpec reprezentuje súbor so zdrojovým kódom, výška stĺpca odpovedá veľkosti súboru. Každý riadok v stĺpcoch reprezentuje jeden riadok kódu. Farba je mapovaná na počet volaní (čím červenšia - tým viac volaní), ale môže reprezentovať aj iné vlastnosti kódu.



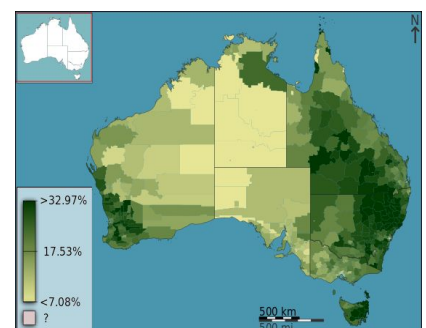
Jigsaw - reprezentuje vzťahy medzi objektami dokumentov a dokumentami. Podporuje *calendar view* (kedy sú objekty dokumentov umiestnené do kalendára na základe dátumu, kedy boli dané entity v texte identifikované). Tiež podporuje vizualizáciu vzťahov pomocou grafov alebo zoznamov. Tieto vizualizácie sú veľmi interaktívne.



14. Choroplet's maps

Je to tématická mapa. Vizualizuje ordinálne plošné dáta. Plocha je zafarbená (či vyplnená textúrou/vzorom) podľa hodnoty nejakej štatistickej premennej, na ktorú je vizualizácia zameraná, napr. Hustota obyvateľstva. Takéto mapy prezentujú nejaké plošné javy v podobne tieňovaných polygónov. Pri generovaní takýchto máp sa väčšinou najprv robí normalizácia dát a farebné a šedotónne mapovanie.

Problémom Choropletových máp je, že najzaujímavejšie hodnoty sú často koncentrované v husto zastúpených oblastiach s malými a ťažko rozpoznateľnými polygónmi, a menej zaujímavé hodnoty sú rozprestreté na veľkých plochách vo väčších polygónnoch, ktoré sú tým pádom opticky dominantnejšie.



16. Effectiveness

Efektivita je jedna z dvoch základných metrík vizualizácii (ako "dobrá" vizualizácia je). Efektívna vizualizácia je "lacná" s ohľadom na čas. Nepotrebuje veľa času na interpretáciu ani na renderovanie. Jednoduchá vizualizácia, ktorá má malý vstupný dataset, spotrebuje tak málo render času, že je v podstate pri výpočte efektivity zanedbateľný. S rastúcou zložitou vizualizácie a/alebo veľkosti datasetu rastie aj čas potrebný na interpretáciu, aj čas potrebný na renderovanie, a teda oba musia byť brané do úvahy vo výpočte efektivity.

Efektivitu môžeme vyjadriť ako pomer m_{eff} :

$$m_{\text{eff}} = 1 / (1 + \text{interpretTime} + \text{renderTime})$$

Čím bližšie je m_{eff} k 0, tým viac času je potrebného na interpretáciu a render, a tým je vizualizácia menej efektívna.

Čím bližšie je m_{eff} k 1, tým menej času je potrebného na interpretáciu a render, a tým je vizualizácia efektívnejšia.

17. Expressiveness

Expresivita je druhá základná vizualizačná metrika. Expresívna vizualizácia poskytuje presne a len tú informáciu, ktorú mienime poskytnúť, nič viac a nič menej.

Môžeme ju definovať ako pomer, m_{exp} , informácie, ktorú vizualizácia predáva, a informácie, ktorú sme chceli, aby vizualizácia podávala. Môže teda nadobúdať hodnoty od 0 vyššie.

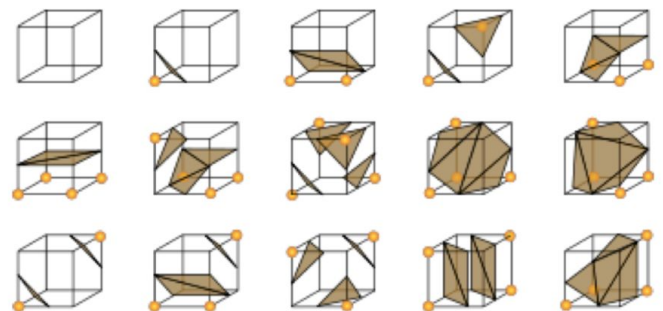
Ak je $m_{\text{exp}} == 1$, je to ideálna situácia, naša vizualizácia podáva presne tú informáciu, ktorú má.

Ak je $m_{\text{exp}} < 1$, podáva vizualizácia menej informácii, než má.

Ak je $m_{\text{exp}} > 1$, podáva vizualizácia viac informácii, než má. Toto môže spôsobiť occlusion informácie, ktoré chceme v skutočnosti zobrazíť.

18. Marching cubes

Algoritmus používaný na renderovanie izopovrchov objemových dát. Voxely sú definované ako kocky s hodnotami v jej ôsmich rohoch. Ak jeden alebo viac rohov má hodnotu nižšiu ako je užívateľom zadefinovaná izohodnota a jeden alebo viac rohov vyššiu hodnotu, potom voxel nejakým spôsobom prispieva ku izopovrchu. Pomocou určovania, ktoré hrany kocky sú pretnuté izopovrchom, tvoríme trojuholníkové vzorky zo všetkých kociek na hranici izopovrchu. Výsledným spojením získame finálnu reprezentáciu povrchu. Na obrázku sú možné pretnutia izopovrchu s kockami - je presne týchto 15 unikátnych možností.



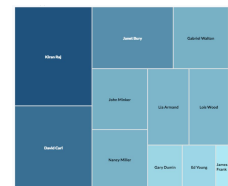
19. Polar coordinates - describe + give an example

Polárna sústava súradníc je sústava súradníc v rovine, ktorá určuje polohu bodu: vzdialenosťou od začiatku sústavy súradníc a uhlom spojnice bodu a začiatku, od začiatku zvolenej osi ležiacej v rovine (proti smeru hodinových ručičiek); najčastejšie jej zodpovedá os x karteziánskych súradníc.

Polárna sústava súradníc je vhodná v takých prípadoch pohybu, pri ktorých sa nemení vzdialenosť telesa od jedného bodu (počiatku súradníc); napríklad pri pohybe po kružnici (prípadne sa táto vzdialenosť mení s určitou jednoduchou závislosťou).

20. Treemap + draw an example

Zobrazuje hierarchické dáta. Obdĺžnik je rekurzívne delený do segmentov, pričom sa strieda horizontálne a vertikálne rozdelenie. Jednotlivé segmenty reprezentujú koncové uzle stromu.



21. Normalization

Často je nutné dáta pred vizualizáciou znormalizovať do nejakého vhodnejšieho intervalu z ľubovoľného pôvodného rozsahu, väčšinou na rozsah [0.0 , 1.0]. Dáta normalizujeme tak, že nájdeme maximálnu a minimálnu hodnotu a každú hodnotu prevedieme formulou:

$$\text{nováHodnota} = (\text{staráHodnota} - \text{min}) / (\text{max} - \text{min})$$

výsledok je hodnota v rozsahu 0.0 až 1.0.

Normalizácia môže taktiež zahŕňať aj orezanie podľa hraničných hodnôt, kedy hodnoty presahujúce prah sú uzavreté na tento prah (je im priradená hraničná hodnota).

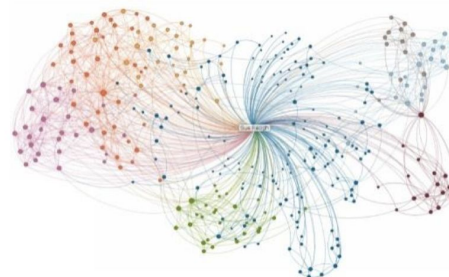
22. Visualization of multidimensional data - describe 3 techniques and sketch pictures

Paralelné súradnice - [popísané vyššie 5.](#)

Tree maps - [popísané vyššie 20.](#)

Siete

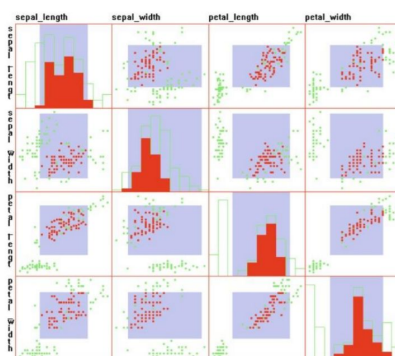
Okrem samotných dát treba vizualizovať aj štruktúru siete. Sieť sa skladá z uzlov = dáta a hrán = vzťahy medzi dátami. Narozdiel od stromov môžu hrany medzi uzlami obsahovať cykly. Pri návrhu siete by sme sa mali sústreďovať na splnenie nasledujúcich kritérií: Minimálne prieniky hrán, Minimálne dĺžky hrán, Minimálne ohýbanie hrán - kvôli čitateľnosti vizualizácie.



23. Matrix scatterplot

Jedná sa o násobné zobrazenie niekoľkých grafov zobrazujúcich rôzne dimenzie vstupných dát. Najčastejšie sa používa práve matrix scatterplot - matica bodových grafov.

Skladá sa z mriežky obsahujúcej bodové grafy, má N^2 buniek, kde N je počet dimenzií. Každá dvojica dimenzií je vykreslená dvakrát - líši sa orotovaním grafu o 90 stupňov. Matica je symetrická podľa hlavnej diagonály. Miesta na tejto diagonále, ktoré by mali zobrazovať premennú v danej dimenzii samu so sebou, sa často používajú pre podanie informácie o dimenziách v odpovedajúcom rade/stĺpci alebo na vykreslenie histogramu danej dimenzie.



24. Geopriestorové vizualizácie

Geopriestorové dáta popisujú objekty alebo javy vyskytujúce sa v reálnom svete. Majú špeciálne vlastnosti a na ich vizualizáciu používame špeciálne metódy. Táto oblasť je široko pokrytá oblasťou GIS (geografických informačných systémov) a kartografiou. Priestorové dáta sú vizualizované priamočiara - sú mapované na dve dimenzie výstupného zariadenia, čím dosiahneme zobrazenie mapy.

Mapy sú vlastne sadami bodov, čiar, a plôch, definovanými v súradnicovom systéme a pomocou ďalších "nepriestorových" atribútov.

Priestorové javy môžeme, z definície, rozdeliť na:

- Bodové javy - Na ich udanie stačí dvojica (zemepisná dĺžka a šírka). Budovy, mestá, ...
- Čiarové javy - majú nejakú dĺžku (teda jednu dimenziu), šírka je daná implicitne. Na ich udanie stačí sada dvojíc (zemepisná dĺžka a šírka). Cesty, hranice, ...
- Plošné (area) javy - majú dĺžku a šírku (dve dimenzie). Na ich udanie stačí sada dvojíc (zemepisná dĺžka a šírka), ktoré sú uzavreté v danom regióne.
- Povrchové (surface) javy - majú dĺžku, šírku, a výšku (majú 2.5 dimenzií). Môžu byť špecifikované sadou vektorov zložených zo zemepisnej dĺžky a šírka a z výšky.

Každá z dvojíc (zemepisná dĺžka a šírka) môže mať asociované ďalšie atribúty.

Existuje množstvo rôznych typov máp. Líšia sa v tom, akú informáciu zobrazujú, ako ju zobrazujú, v merítku, vo využití, ... Jedným z dôležitých faktorov pri vytváraní mapy je, aké veľké územie má mapa zobrazovať. Pokiaľ totiž tvoríme mapu zobrazujúcu celú Zem (alebo jej veľkú časť), narážame na problém projekcie.

Zemeguľa, ako 3D objekt prevažne guľovitého tvaru, nemôže byť presne, bez chýb, zobrazená na iný povrch. Takýmto zobrazeniam hovoríme projekcie. Existuje mnoho druhov projekcií s rôznymi chybami, ktoré však môžu závisieť aj na účele danej projekcie. Niekedy je dôležitejšie zachovať tvar, aj keď je skreslená veľkosť nejakej plochy. Inokedy je zas potrebné čo najpresnejšie zachovať veľkosť a to na úkor korektného tvaru.

Projekcie môžeme klasifikovať podľa typu povrchu, na ktorý zobrazujeme, na:

- Valcové
- Rovinné
- Kužeľové

Vizualizácia priestorových dát

Bodové mapy - zobrazenia množstva jednotlivých záznamov na mape, ukazuje to ich vzťahy, clustering, frekvenciu... Používajú sa symboly, ktoré môžu mať rôzne vizuálne vlastnosti na zdôraznenie či pridanie dodatočných informácií.

Čiarové mapy - zobrazenia ciest, sietí, ... Opäť rôzne vizuálne atribúty na zdôraznenie či pridanie dodatočných informácií.

Mapy toku - flow maps - inšpirované grafovými algoritmami, minimalizujú pretínanie hrán a deformáciu pozíc uhlov, zachovávajú ich relatívnu pozíciu. Hierarchické klastrovanie podľa pozície uzlov a tokov medzi nimi. Zhlukovanie hrán pre čistejšie grafy.

Plošné mapy - Choropletové mapy ([viď 15.](#)), kartogramy ([viď 10](#)), dasymetrické mapy (wtf), or, you know, klasické tématické mapy

Map labeling nie je vôbec jednoduchý!!.

Vizuálne premenné pre priestorové dáta

- Veľkosť - symbolov, šírka čiar, ...
- Tvar - symbolov alebo vzorov v čiarach a plochách
- Jas - symbolov, čiar a plôch
- Farba - symbolov čiar a plôch
- Orientácia - symbolov alebo vzorov v čiarach a plochách
- Textúra - rozmiestnenie vzorov v symboloch, čiarach a plochách
- Perspektívna výška - perspektívny 3D pohľad na javy, kde sú dátové hodnoty mapované na perspektívnu výšku jednotlivých bodov, čiar, alebo plôch
- Rozmiestnenie, usporiadanie (arrangement) - rozmiestnenie vzorov v jednotlivých symboloch, vzorov bodiek a čiarok čiarových javov, vzorov pravidelného(náhodného) rozmiestnenia symbolov pre plošné javy

	Size	Shape	Brightness	Color	Orientation	Spacing	Perspective height	Arrangement
Point								
Linear								
Areal								

25. Juxtaposition and superimposition

Juxtapositioning je umiestnenie vizualizácií viacerých premenných nejakej dátovej sady **vedľa seba** tak, aby mal user prehľad o všetkých týchto premenných.

Superimpositioning je umiestnenie vizualizácií viacerých premenných nejakej dátovej sady **na seba** tak, aby mal user prehľad o všetkých týchto premenných. Navyše takto môže vidieť rozdiely medzi týmito vlastnosťami, napr. rôzny vývoj hodnôt daných premenných.



Juxtapositioning vľavo, superimpositioning vpravo

26. Volumetric data visualization (+ marching cubes [vid' 18](#))

Pri väčšine techník vizualizácie objemových dát hrá veľkú úlohu prevzorkovanie.

Plátovanie (slicing)

- Využíva orezávacie roviny, ktoré môžu byť osovo zarovnané, alebo mať ľubovoľnú orientáciu. Výsledkom je 2D plát vystrihnutý z dát, ktorý môžeme následne zobrazit' pomocou niektorej z 1D alebo 2D vizualizačných techník.

Izopovrchy

- User špecifikuje vstupné parametre, podľa ktorých je generovaný popis povrchu dátovej sady, a je vizualizovaný pomocou jednej z techník vizualizácie explicitného povrchu
- Marching cubes

Priame renderovanie objemu

- Dve metódy:
 - Prvá založená na vrhnutí paprsku do scény obsahujúcej objemové teleso, a následný výpočet hodnôt v jednotlivých pixeloch na základe voxelov, ktoré paprsok prešiel
 - Druhá premietne každý voxel na projekčnú rovinu použitím niektorej z metód akumulácie efektov voxelov na pixely

27. Multivariate data visualization

Dáta, ktoré sa skladajú z rôznych typov atribútov (napr. Dátová sada, ktorá drží informácie o váhe w (desatinné číslo), výške h (celé číslo) a veľkosti topánok (celé číslo) náhodnej vzorky osôb).

Bodové dáta - scatterplots

So zvyšujúcou dimenzionalitou vstupných dát sa vizuálna analýza skladá z:

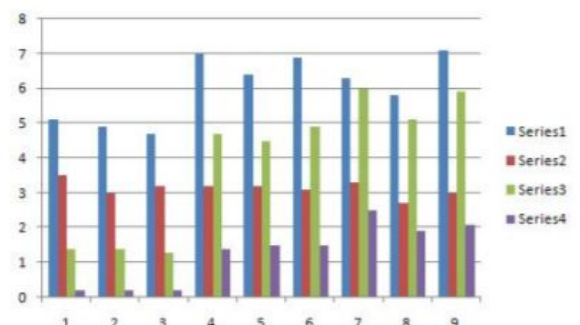
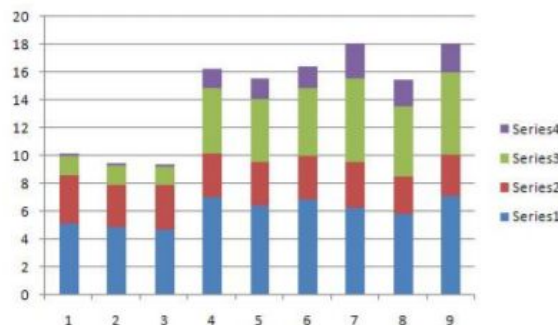
- Hľadania podmnožiny vstupných dimenzií - dimension subsetting - povolíme userovi zobrazit' len nejakú podmnožinu vstupných dimenzií. Pre nájdenie tejto podmnožiny použijeme algoritmy na hľadanie dimenzií obsahujúcich pre danú úlohu najužitočnejšie informácie
- Dimension reduction - PCA, MDS
- Ukotvenie dimenzie - dimension embedding - mapovanie dimenzií na ďalšie grafické atribúty, ako je farba, veľkosť, tvar
- Násobné zobrazenie - multiple displays - juxtaposition/superimposition (matrix scatterplots)

Techniky pre čiarové dáta

- Čiarové grafy, prekrývanie, small multiples, sparklines, umiestnenie relatívne k predchádzajúcemu záznamu na osy y
- Paralelné súradnice
- Radiálna osa
 - Radar
 - Polárne grafy (polárne súradnice)
 - Kruhové stĺpcové/plošné diagramy
 - Hviezdicové grady

Techniky pre plošné dáta

- Stĺpcové diagramy/histogramy



- Cityscapes - 3D stĺpce umiestnené na mape - časté prekrývanie
- Tabuľkové zobrazenie
 - Heatmapy
 - zobrazenie tabuľky záznamov za použitia farby namiesto textu. Hodnoty sú mapované do normalizovaného farebného priestoru

- Survey plots - prehľadové grafy
 - Namiesto s farbou buniek pracujeme s ich veľkosťou, a zarovnávame ich na stred

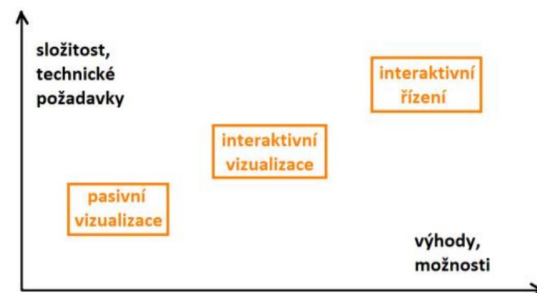
28. Passive visualization vs Interactive visualization vs Interactive steering

Proces vizualizácie sa skladá z 3 základných fáz:

- Generovanie dát
- Vizualizácia dát
- Interakcia s užívateľom

V kontexte týchto troch fáz môžeme hovoriť o 3 základných typoch vizualizácie:

- Pasívna vizualizácia
 - Všetky tri fázy sú striktné oddelené
 - User si len prezerá výsledné vizualizácie, bez akéhokoľvek zapojenia do prvých dvoch fáz
- Interaktívna vizualizácia
 - Oddelená fáza generovaných dát
 - User si môže dáta vo vizualizácii mierne upravovať, prípadne meniť vlastnosti vizualizácie a rôznu s ňou interagovať
- Interactive steering (riadenie)
 - User môže ovplyvňovať všetky tri fázy
 - Dáta sú generované on-the-fly, priamo spracovávané vo vizualizačnej fáze
 - Tá umožňuje "real-time" interaktívny náhľad na dáta



29. Selection vs. filtering. What is fish-eye? When is it used? What are the problems?

Selection

User izoluje podmnožinu komponent pre zobrazenie, tie ďalej podliehajú ďalším operáciám, ako je zvýraznenie, maskovanie, presunutie do oblasti záujmu... Existujú rôzne spôsoby výberu, je ale skoro vždy robený priamo - označujeme objekty klikaním, kreslením, obdĺžnikovým výberom, ...

Jednou z hlavných tém je granularita výberu. Malo by kliknutie na nejakú vec vybrať len tú vec, alebo celý objekt, do ktorého hierarchicky patrí?

Filtering

Redukuje počet dát, ktoré majú byť zobrazené, pomocou rôznych obmedzení, ktoré špecifikujú, ktoré dáta budú zachované a ktoré odstránené. Pre určenie rozsahu záujmu sa napr. používajú slidere, pri ktorých použitie je vizualizácia hneď aktualizovaná, aby reflektovala zmeny vyvolané userom. Inou metódou je výber položiek, ktoré chceme zachovať, či naopak, schovať.

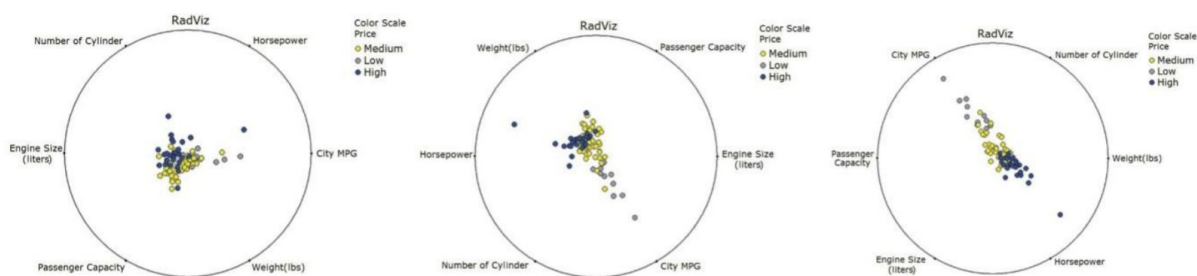
Zásadným rozdielom medzi selekciou a filtráciou je, že filtrácia je robená nepriamo - v dialógových oknách, nastaveniach, rozhraniach, často pred samotným zobrazením dát (aby sme sa vyhli zobrazeniu príliš veľkého množstvu dát).

Fish-eye - [vid' vyššie](#)

30. RadViz, what is the interpretation of a point in this visualization, interaction, what is the result of interacting with it? Sketch a picture. What is Vectorized RadViz?

RadViz

Je projekcia viacerých dimenzií do dvoch, a to konkrétne do kružnice. Na kružnici sú kotvy - atribúty, a vnútri kružnice sú záznamy. Záznamy sú reprezentované bodmi a sú umiestnené podľa toho, akú majú hodnotu jednotlivých atribútov. "Pružiny", ktorými sú záznamy ukotvené o kotvy, musia po sčítaní dať 0. Dochádza síce k prekryvaniu viacerých záznamov v kružnici, ale to sa deje vždy pri projekcii. RadViz je však zamýšľaný ako veľmi interaktívna vizualizácia, kde si môže user meniť poradie atribútov na kružnici, a nájsť tak pre seba ideálne zobrazenie.

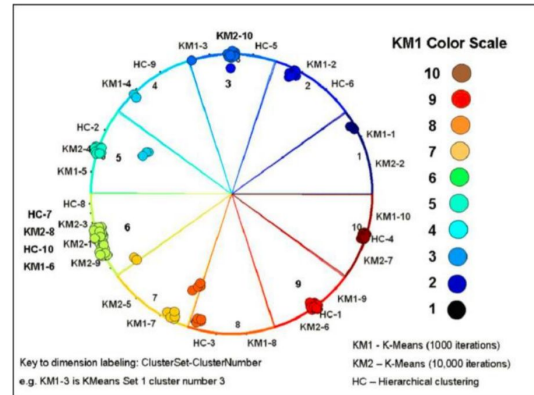


Vectorized RadViz

Zo študijných materiálov:

“Konstruuje násobné dimenze z jednotlivých dimenzí. Jako příklad si uveďme rozložení dimenze reprezentující počet válců motoru auta do pěti nových dimenzí: první obsahuje pouze 1 nebo 2 válce, druhá 3 nebo 4 válce, třetí 5 nebo 6 válců, čtvrtá 7 válců a pátá 8 válců. Počet nových dimenzí může být určen algoritmicky nebo ručně. Tento proces je velmi podobný metodě třídění dat do košů (např. podle nízké, střední a vysoké ceny automobilů).

Každá původní dimenze je tedy reprezentována vektorem nových dimenzí, kdy každá nová souřadnice v takovém vektoru nabývá hodnoty 0 nebo 1 podle toho, zda daný záznam obsahuje hodnotu odpovídající této dimenzi nebo ne. Proto pro každý takový záznam obsahuje každý nový vektor právě jednu dimenzi obsahující hodnotu 1 a všechny ostatní mají hodnotu 0. Příklad VRV je ukázán na následujícím obrázku.“



31. Dostanete obrázok vizualizácie, ktorá používa 2 rôzne techniky. Popíšte kompozíciu, aké dáta vizualizuje, ako by mohla byť vylepšená.
32. Dostane obrázok vizualizácie. Popíšte princíp a na čo sa používa.
33. Dostanete obrázok vizualizácie. Diskutujte o pozitívach a negatívach zvoleného prístupu. Čo by sa dalo vylepšiť? Má vizualizácia závažné nedostatky? Navrhnite vlastnú (vhodnejšiu) vizualizáciu pre dané dáta.