Appendix A. Calculating entropic measures of a map

Appendix A.1. Calculating Shannon's entropy

Given a source S with m events and probabilities $p_1, p_2, \ldots, p_m$ such that $\sum_{i=1}^{m} p_i = 1$, the entropy H of the source S is computed as $H(S) = -\sum_{i=1}^{m} p_i \log_2 p_i$, where $0 \log 0 = 0$. The subscript of 2 under the log symbol signifies bits units. The entropy of two independent sources S and T is given by $H(S,T) = H(S) + H(T)$. If the sources are dependent, the conditional entropy is used to obtain $H(S,T)$, as follows $H(S,T) = H(S) + H(T|S) = H(T) + H(S|T)$ [19].

Appendix A.2. Using Shannon's entropy to estimate coding complexity

As noted in the main text, two major sources of coding complexity considered here are source A, which relates to the events of the alphabets in each column $\boldsymbol{a_j}$ of the map, for $j : 1, \ldots, n$, and source B, which relates to the events of the combinations of the rows (or codes) of the same map. Shannon's entropy of column $\boldsymbol{a_j}$, $H(\boldsymbol{a_j})$, is given by $H(\boldsymbol{a_j}) = -\sum_{i=1}^{k_j} p_{ij} \log_2 p_{ij}$ where $k_j \leq m$ is the number of unique alphabets in column $\boldsymbol{a_j}$ and p_{ij} is the probability of alphabet i in position j. From Equation 2, it was indicated that $H(A) = \sum_{j=1}^{n} H(\boldsymbol{a_j}) = -\sum_{j=1}^{n} \sum_{i=1}^{k_j} p_{ij} \log_2 p_{ij}$. A more refined measure of $H(A)$ is possible to reflect the fact that the position of an alphabet in a code may carry a different weight of a coding error. For example, in ICD-10, a coding error where the first character is incorrect is typically much worse than a coding error where the last character is wrong since the first alphabet tends to serve as a root node for the classification of clinical concepts. Accordingly, a weighing scheme can be devised to account for the relative influence of the position of an alphabet in a code. If w_j were the weight of position j in the code, the resulting weighted average entropy of source A, $H(\bar{A})$, may look like this:

$$H(\bar{A}) = \frac{\sum_{j=1}^{n} \log_2 \left(\prod_{i=1}^{k_j} p_{ij}^{-p_{ij}} \right)^{w_j}}{\sum_{j=1}^{n} w_j} \tag{A.1}$$

where, as before, $k_j \leq m$ is the number of unique alphabets in column $\boldsymbol{a_j}$ and p_{ij} is the probability of alphabet i in position j, for $j : 1, \ldots, n$. The proof of Equation A.1 follows from a logarithmic rules of $\log(x.y) = \log(x) + \log(y)$ and $a. \log(x) = log(x)^a$. Hence, it follows that $-\sum_{j=1}^{n} \sum_{i=1}^{k_j} p_{ij} \log p_{ij} = -\sum_{j=1}^{n} \sum_{i=1}^{k_j} \log p_{ij}^{-p_{ij}} = \sum_{j=1}^{n} \log \left(\prod_{i=i}^{k_j} p_{ij}^{-p_{ij}} \right)$. After weighing each column, it is evident that $\sum_{j=1}^{n} w_j \cdot \log \left(\prod_{i=i}^{k_j} p_{ij}^{-p_{ij}} \right)$ is equivalent to $\sum_{j=1}^{n} \log \left(\prod_{i=1}^{k_j} p_{ij}^{-p_{ij}} \right)^{w_j}$. The denominator in Equation A.1 allows for the calculation of the average of the weighted entropy.

As for source B, it was indicated in Equation 3 that $H(B) = \log(v)$. The proof of this equation follows from the fact that, under the assumption of a uniform distribution, each valid representation is equally likely, with probability $1/v$. Then by Shannon's entropy, $H(B) = -\sum_{j=1}^{v} \frac{1}{v} \log \left(\frac{1}{v} \right) = -\frac{v}{v}(\log 1 - \log(v)) = log(v)$. If all m number of candidate codes in a map have a one-to-one relationship with code x, then all these candidate codes are considered stand-alone and don't have to be combined to form a valid representation of code x. In such case, the number of valid representations v equals the number of stand-alone m_0, which also equals the number of candidate codes m. This means that:

$$H(B) = \log(v) \equiv \log(m) \equiv \log(m_0) \tag{A.2}$$

Equation A.2 is comparable to a case of the UR measure in Chen et al. [18]. If a map includes some stand-alone codes and other codes that need to be combined under different scenarios to make valid representations of code x, then v is obtained by:

$$v = m_0 + \sum_{i=1}^{s}[(m_1)(m_2)\dots(m_{m-m_0})]_i \equiv m_0 + \sum_{i=1}^{s}\prod_{j=1}^{m-m_0} m_{ij} \tag{A.3}$$

where i represents scenario i out of s total number of scenarios. Here, m_0 is again the number of stand-alone codes and $m - m_0$ is the number of codes that must be combined in sequential order of their index, as a set, to represent the old code x. That is, for a given scenario i, m_{i1} is the number of codes that must be sequenced first, followed by m_{i2}, the total number of codes that must be sequenced second followed by m_{i3}, the total number of codes that must be sequenced third, and so on until $m_{i(m-m_0)}$. If $m = m_0$, then, as indicated in Equation A.2, $v = m_0$. The justification of Equation A.3 comes from the fact that if a map has some stand-alone codes, then there are $\binom{m_0}{1} = \frac{m_0!}{1!(m_0-1)!} = m_0$ possibilities of choosing one stand-alone code at random. If a map has at least one scenario, then for each scenario i, there are $\binom{m_{i1}}{1}.\binom{m_{i2}}{1}\dots\binom{m_{i(m-m_0)}}{1} = (m_{i1})(m_{i2})\dots(m_{i(m-m_0)})$ possibilities of choosing one sequence of codes from m_{i1} to $m_{i(m-m_0)}$. Since stand-alone codes don't have to be combined with any other codes, then, for a map with a single scenario, the number of valid representations of code x is given by $v = m_0 + (m_1)(m_2)\dots(m_{m-m_0})$ (review the example in Figure 1). If a map had more than one scenario, the total number of valid representations of code x would follow Equation A.3.

Appendix A.3. Normalizing the entropic measures

Assuming that $H(A) \equiv \alpha$ and $H(B) \equiv \beta$, the normalized entropic scores can be obtained this way:

$$Z(\alpha) = \frac{\alpha - \bar{\alpha}}{var(\alpha)} \tag{A.4}$$

$$Z(\beta) = \frac{\beta - \bar{\beta}}{var(\beta)} \tag{A.5}$$

where $\bar{\alpha}$ is the average of α measures from all maps and $\bar{\beta}$ is the average of β measures from all maps. The symbol $var()$ signifies the variance function.

Appendix B. Top 30 maps in the forward mapping from ICD-9-CM to ICD-10-CM

Table B.4: Top 30 maps in the forward mapping from ICD-9-CM to ICD-10-CM, ranked by their sum of $Z(\alpha)$, $Z(\beta)$, and $Z(UR)$, from most to least total score. The map id and description correspond to ICD-9-CM code and description, respectively.

Map	Map description	$Z(\alpha)$	$Z(\beta)$	$Z(UR)$	Total score
V5412	Aftercare for healing traumatic fracture of lower arm	7.99	12.72	12.15	32.87
V5416	Aftercare for healing traumatic fracture of lower leg	7.95	12.47	11.90	32.31
V5411	Aftercare for healing traumatic fracture of upper arm	6.74	10.80	10.31	27.86
99529	Unspecified adverse effect of other drug, medicinal and biological substance	7.44	10.19	9.72	27.36
V5413	Aftercare for healing traumatic fracture of hip	7.14	10.19	9.72	27.05
V5417	Aftercare for healing traumatic fracture of vertebrae	6.78	10.00	9.54	26.32
V5415	Aftercare for healing traumatic fracture of upper leg	7.09	9.80	9.35	26.23
9895	Toxic effect of venom	6.12	10.10	9.63	25.85
99811	Hemorrhage complicating a procedure	10.01	8.08	7.70	25.78
99812	Hematoma complicating a procedure	10.01	8.08	7.70	25.78
V5419	Aftercare for healing traumatic fracture of other bone	7.73	9.11	8.69	25.53
29289	Other specified drug-induced mental disorders	6.43	9.34	8.91	24.68
9982	Accidental puncture or laceration during a procedure, not elsewhere classified	9.94	6.14	5.84	21.92
73382	Nonunion of fracture	6.45	6.90	6.56	19.91
9050	Late effect of fracture of skull and face bones	5.45	6.83	6.50	18.78
9947	Asphyxiation and strangulation	4.53	7.25	6.90	18.68
98989	Toxic effect of other substance, chiefly nonmedicinal as to source, not elsewhere classified	5.14	6.47	6.16	17.77
24980	Secondary diabetes mellitus with other specified manifestations, not stated as uncontrolled, or unspecified	4.93	6.47	6.16	17.57
9880	Toxic effect of fish and shellfish eaten as food	5.19	6.23	5.92	17.34
9823	Toxic effect of other chlorinated hydrocarbon solvents	4.36	6.55	6.23	17.14
9063	Late effect of contusion	6.13	5.63	5.35	17.10
8065	Open fracture of lumbar spine with spinal cord injury	5.13	5.63	5.92	16.68
9057	Late effect of sprain and strain without mention of tendon injury	6.16	5.38	5.11	16.64
E959	Late effects of self-inflicted injury	4.82	5.85	5.56	16.22
E9990	Late effect of injury due to war operations	4.81	5.85	5.56	16.21
64131	Antepartum hemorrhage associated with coagulation defects, delivered, with or without mention of antepartum condition	3.72	6.31	6.01	16.04
8064	Closed fracture of lumbar spine with spinal cord injury	4.49	5.63	5.92	16.03
9064	Late effect of crushing	5.50	5.38	5.11	15.98
986	Toxic effect of carbon monoxide	4.56	5.85	5.56	15.97
73395	Stress fracture of other bone	5.38	5.38	5.11	15.87

Appendix C. Outliers maps of $Z(\alpha)$ where $Z(\alpha) > 2.7$ (about top 1% of the maps)

Table C.5: Outlier maps arranged in the decreasing order of the $Z(\alpha)$ scores. As before, m is the number of candidate codes in a map whereas v is the number of valid representations in a map. The map id and description correspond to ICD-9-CM Vol. 3 code and description, respectively.

Map	Map description	m	v	$Z(\alpha)$
3929	other (peripheral) vascular shunt or bypass	1191	1191	4.87
8605	incision with removal of foreign body or device from skin and subcutaneous tissue	415	415	4.06
8609	other incision of skin and subcutaneous tissue	335	335	4.00
3950	angioplasty of other non-coronary vessel(s)	1196	1196	3.68
0109	other cranial puncture	34	34	3.68
843	revision of amputation stump	349	349	3.60
9301	functional evaluation	148	148	3.57
0404	other incision of cranial and peripheral nerves	327	327	3.47
9788	removal of external immobilization device	98	98	3.42
3979	other endovascular procedures on other vessels	689	689	3.41
9223	radioisotopic teleradiotherapy	768	768	3.38
046	transposition of cranial and peripheral nerves	378	378	3.37
8382	graft of muscle or fascia	440	440	3.35
0124	other craniotomy	111	111	3.29
3926	other intra-abdominal vascular shunt or bypass	720	720	3.29
409	other operations on lymphatic structures	510	510	3.27
8196	other repair of joint	313	313	3.26
9227	implantation or insertion of radioactive elements	268	268	3.21
3897	central venous catheter placement with guidance	48	320	3.15
0474	other anastomosis of cranial or peripheral nerve	350	350	3.15
3821	biopsy of blood vessel	699	699	3.08
8120	arthrodesis of unspecified joint	582	582	3.06
3958	repair of blood vessel with unspecified type of patch graft	421	421	3.05
8604	other incision with drainage of skin and subcutaneous tissue	281	281	3.01
3348	other repair and plastic operations on bronchus	288	288	3.00
8129	arthrodesis of other specified joints	552	552	3.00
2103	control of epistaxis by cauterization (and packing)	10	8	3.00
9649	other genitourinary instillation	5	5	2.97
3956	repair of blood vessel with tissue patch graft	402	402	2.95
8100	spinal fusion, not otherwise specified	282	282	2.88
8683	size reduction plastic operation	340	340	2.84
9205	cardiovascular and hematopoietic scan and radioisotope function study	54	54	2.83
0309	other exploration and decompression of spinal canal	70	70	2.78
8080	other local excision or destruction of lesion of joint, unspecified site	345	345	2.77
8040	division of joint capsule, ligament, or cartilage, unspecified site	210	210	2.74
3925	aorta-iliac-femoral bypass	320	320	2.73
3805	incision of vessel, other thoracic vessels	78	78	2.71

Appendix D. Outliers maps of $Z(\beta)$ where $Z(\beta) > 2.85$ (about top 1% of the maps)

Table D.6: Outlier maps arranged in the decreasing order of the $Z(\beta)$ scores. As before, m is the number of candidate codes in a map whereas v is the number of valid representations in a map. The UR measure (obtained by $\log_2(m)$) can directly be compared to $H(B)$ (obtained by $\log_2(v)$). When $UR < H(B)$, UR has underestimated the expected coding complexity. When $UR > H(B)$, UR has overestimated the expected coding complexity. Otherwise, the measures are equal. The map id and description correspond to ICD-9-CM code and description, respectively.

Map	Map description	$H(B)$	UR	m	v	$\boldsymbol{Z(\beta)}$
3473	Closure of other fistula of thorax	10.95	7.92	243	1977	4.26
3950	Angioplasty of other non-coronary vessel(s)	10.22	10.22	1196	1196	3.88
3929	Other (peripheral) vascular shunt or bypass	10.22	10.22	1191	1191	3.88
304	Radical laryngectomy	9.61	5.81	56	784	3.57
9223	Radioisotopic teleradiotherapy	9.58	9.58	768	768	3.55
3926	Other intra-abdominal vascular shunt or bypass	9.49	9.49	720	720	3.50
3821	Biopsy of blood vessel	9.45	9.45	699	699	3.48
3979	Other endovascular procedures on other vessels	9.43	9.43	689	689	3.47
8120	Arthrodesis of unspecified joint	9.18	9.18	582	582	3.34
5684	Closure of other fistula of ureter	9.13	6.13	70	560	3.31
8129	Arthrodesis of other specified joints	9.11	9.11	552	552	3.30
409	Other operations on lymphatic structures	8.99	8.99	510	510	3.24
8687	Fat graft of skin and subcutaneous tissue	8.98	6.04	66	506	3.24
5783	Repair of fistula involving bladder and intestine	8.98	6.13	70	505	3.23
3342	Closure of bronchial fistula	8.78	7.03	131	440	3.13
5784	Repair of other fistula of bladder	8.78	6.19	73	440	3.13
8382	Graft of muscle or fascia	8.78	8.78	440	440	3.13
3794	Implantation or replacement of automatic cardioverter/defibrillator, total system [aicd]	8.75	5.25	38	432	3.12
3958	Repair of blood vessel with unspecified type of patch graft	8.72	8.72	421	421	3.10
8605	Incision with removal of foreign body or device from skin and subcutaneous tissue	8.70	8.70	415	415	3.09
3956	Repair of blood vessel with tissue patch graft	8.65	8.65	402	402	3.06
046	Transposition of cranial and peripheral nerves	8.56	8.56	378	378	3.02
8663	Full-thickness skin graft to other sites	8.56	5.81	56	378	3.02
3991	Freeing of vessel	8.49	8.49	360	360	2.98
3957	Repair of blood vessel with synthetic patch graft	8.48	8.48	358	358	2.98
3951	Clipping of aneurysm	8.48	8.48	357	357	2.97
0474	Other anastomosis of cranial or peripheral nerve	8.45	8.45	350	350	2.96
843	Revision of amputation stump	8.45	8.45	349	349	2.96
3959	Other repair of vessel	8.43	8.43	346	346	2.95
8080	Other local excision or destruction of lesion of joint, unspecified site	8.43	8.43	345	345	2.95
8683	Size reduction plastic operation	8.41	8.41	340	340	2.94
8609	Other incision of skin and subcutaneous tissue	8.39	8.39	335	335	2.93
0404	Other incision of cranial and peripheral nerves	8.35	8.35	327	327	2.91
3897	Central venous catheter placement with guidance	8.32	5.58	48	320	2.89
3925	Aorta-iliac-femoral bypass	8.32	8.32	320	320	2.89
8196	Other repair of joint	8.29	8.29	313	313	2.88
8381	Tendon graft	8.29	8.29	312	312	2.87
8383	Tendon pulley reconstruction other than hand	8.29	8.29	312	312	2.87

Appendix E. Implementation of Algorithm 4.1 using Python 3.6

To download the gem_i9pcs.txt file used in this code - this file represents the forward mappings from ICD-9-CM Vol.3 to ICD-10-PCS, go to *https://www.cms.gov/Medicare/Coding/ICD10/2015-ICD-10-PCS-and-GEMs*>*2015 General Equivalence Mappings (GEMs) – Procedure Codes and Guide (ZIP)* >*gem_i9pcs.txt*

```python
import pandas as pd
from math import log
from collections import Counter
import numpy as np
##Import forward mappings, ICD-10-CM Vol 3 to ICD-10-PCS. If importing backward mappings, add a padding to icd9 codes so all candidate codes have a fixed length (e.g., use lambda x:x.ljust(7, '0') to have all codes be 7 characters long). Also, for backward mappings, ensure to replace all instances of pcs (in this script) with icd9.
icd10=pd.read_csv(r'path\gem_i9pcs.txt',sep='\s+',names=['icd9','pcs','flag'],converters={'flag': lambda x: str(x),'icd9': lambda x: str(x) })
icd10 = icd10[~icd10.pcs.isin(['NoPCS', 'NoDx'])]#Exclude cases with no match in the target system
##Relevant functions to implement Algorithm 4.1
def slice_code(code,n):
    '''slice a code into axes(e.g., [ABC]-->[A,B,C]) where n is the constant code length in a map'''
    if code =='NoPCS':
        pass
    else:
        return code[n]
def count_flag(code):
    ''' Count the number of possible combinations, given the flag column in icd10 file'''
    b = pd.Series(code).value_counts().sort_index()
    b = pd.Series(b)
    c = list(b.index)
    d = list(b.array)
    g=[]
    for i in b.index:
        g.append (int(str(i)[3]))
    s1 = pd.Series(d)
    s2 = pd.Series(g)
    s3 = pd.concat([s2,s1], axis=1)
    s4 = s3.groupby(0)[1].apply(np.prod)
    s4 = pd.Series(s4)
    s5 = sum(list(s4.array))
    return s5
def entropy(aj):
    '''calculate the entropy of each axis (aj) to obtain H(a_j)'''
    p, lns = Counter(aj), float(len(aj))
    return abs(round(-sum( ((count/lns)*log(count/lns, 2))  for count in p.values()),2))
```

```
def log_function(x):
    '''calculate the entropy, given v (number of possible combinations) to obtain H(B)'''
    if x==0:
        return 0
    else:
        return log (x,2)
##STEP 1: CALCULATE H(A)
#Slice the pcs code
icd10['section']=icd10.pcs.apply(slice_code, n=0)
icd10['system']=icd10.pcs.apply(slice_code, n=1)
icd10['root']=icd10.pcs.apply(slice_code, n=2)
icd10['part']=icd10.pcs.apply(slice_code, n=3)
icd10['approach']=icd10.pcs.apply(slice_code, n=4)
icd10['device']=icd10.pcs.apply(slice_code, n=5)
icd10['qualifier']=icd10.pcs.apply(slice_code, n=6)
#Calculate H(aj)
axis1 = icd10.groupby('icd9')['section'].apply(entropy)
axis2  = icd10.groupby('icd9')['system'].apply(entropy)
axis3  = icd10.groupby('icd9')['root'].apply(entropy)
axis4  = icd10.groupby('icd9')['part'].apply(entropy)
axis5  = icd10.groupby('icd9')['approach'].apply(entropy)
axis6  = icd10.groupby('icd9')['device'].apply(entropy)
axis7  = icd10.groupby('icd9')['qualifier'].apply(entropy)
#Aggreate a dataframe of all column entropic measures
maps = pd.concat([axis1,axis2,axis3,axis4,axis5,axis6,axis7], axis =1)
maps = pd.DataFrame(maps)
maps.columns = ['H(a1)','H(a2)','H(a3)','H(a4)','H(a5)','H(a6)','H(a7)']
#Equal weights for axes: np.array([1,1,1,1,1,1,1]). H(A)=sum(H(aj))
weights = np.array([1,1,1,1,1,1,1])
maps['H(A)'] = np.dot(maps, weights)
##STEP 2: CALCULATE H(B) AND UR
maps['v'] = icd10.groupby('icd9')['flag'].apply(count_flag)
maps['H(B)'] = maps['v'].apply(lambda x: log_function(x))
maps['m'] = icd10.groupby('icd9')['flag'].count()
maps['UR'] = maps['m'].apply(lambda x: log_function(x))
##STEP 3: NORMALIZE ENTROPIC MEASURES
#Normalize H(A)
avg = maps['H(A)'].mean()
var = maps['H(A)'].var()
maps['Z(alpha)'] = maps['H(A)'].apply(lambda x: (x-avg)/np.sqrt(var))
#Normalize H(B)
avg = maps['H(B)'].mean()
var = maps['H(B)'].var()
maps['Z(beta)'] = maps['H(B)'].apply(lambda x: (x-avg)/np.sqrt(var))
#Normalize the UR measure
```

```
avg = maps['UR'].mean()
var = maps['UR'].var()
maps['Z(UR)'] = maps['UR'].apply(lambda x: (x-avg)/np.sqrt(var))
##STEP 4: WEIGH ENTROPIC MEASURES by the historical frequency distribution of clincal
    concepts (if data available)
##STEP 5: RANK MAPS from highest to lowest entropic measure
maps['Zsum'] = maps['Z(alpha)']+maps['Z(beta)']+maps['Z(UR)']
##maps_rankall: rank maps by combining all entropic measures
maps_rankall = maps.sort_values(by=['Zsum'],  ascending=False)
##maps_rank1: rank maps by Z(alpha)
maps_rank1 = maps.sort_values(by=['Z(alpha)'],  ascending=False)
##maps_rank2: rank maps by Z(beta)
maps_rank2 = maps.sort_values(by=['Z(beta)'],  ascending=False)
##maps_rank3: rank maps y Z(UR)
maps_rank3 = maps.sort_values(by=['Z(UR)'],  ascending=False)
```